Supporting Information for

On a closure scheme for chemical master equations

Patrick Smadbeck, Yiannis N. Kaznessis*

Department of Chemical Engineering and Materials Science, University of Minnesota
421 Washington Ave SE, Minneapolis, MN 55455, USA

This file includes:

Section S1 – Theoretical Foundation for Zero-Information Closure

This section explains the theoretical underpinnings of the zero-information moment closure method. The section starts with an outline of zero information closure (ZI-closure) and then provides the algorithm for determining the maximum entropy distribution. It concludes with a simple reversible dimerization example to clarify the determination and implementation of several important equations. Further supplemental sections expand upon this information to describe an iterative ODE solving scheme and the steady-state determination method. The reversible dimerization example is extended in both cases all well.

A closure scheme is used to determine the key relationship between higher and lower-order moments, necessary to close and solve the moment equations:

$$\underline{\mu}' = F\left(\underline{\mu}\right) \tag{S1}$$

In most closure schemes F(μ) is an analytical expression related to a well-known characteristic equation. In the case of ZI-closure, however, the closure scheme refers to the maximum-entropy distribution or most-likely distribution.

For a single component system the information entropy is defined as:

$$H = -\sum_{x=0}^{\infty} p(x) \ln p(x) \tag{S2}$$

For an unconstrained system the resulting maximum-entropy distribution is simply uniform. However, values of the lower-order moments, μ, act as constraints on the system. The result is best solved using a Lagrange multiplier method (here assuming a simple component with M known lower order moments):

$$\Lambda = H - \lambda_0 g_0 - \lambda_1 g_1 - \cdots - \lambda_M g_M \tag{S3}$$

$$\begin{aligned} g_0 &= \sum_{x=0}^{\infty} p(x) - 1 \\ g_1 &= \sum_{x=0}^{\infty} x p(x) - \langle x \rangle \\ &\vdots \\ g_M &= \sum_{x=0}^{\infty} x^M p(x) - \langle x^M \rangle \end{aligned} \tag{S4}$$

Taking Equation S3, the maximum is found by differentiating by p(x) and setting the result to zero:

$$\frac{\partial \Lambda}{\partial p(x)} = -\ln p(x) - 1 - \lambda_0 - \lambda_1 x - \cdots - \lambda_M x^M = 0 \tag{S5}$$

$$p_H(x) = \exp\left(-1 - \lambda_0 - \lambda_1 \mathrm{x} - \cdots - \lambda_\mathrm{M} \mathrm{x}^\mathrm{M}\right) \tag{S6}$$

An analytical expression for the maximum entropy distribution is determined with the same number of parameters as the number of known lower-order moments. The derivation of Equation S6 is easily

extended to multi-component systems and, although we do not explore it here, continuous state-space systems.

Using the lower-order moments, $\underline{\mu}$, the maximum entropy distribution is found by determining the Lagrange parameters, $\underline{\lambda}$. The maximum-entropy moments, denoted with a subscript H, can be determined with the known $p_H(x)$. Take, for example, the determination of $\langle x^m \rangle_H$ for any arbitrarily high *m* in a single component system:

$$\langle x^m \rangle_H = \sum_{x=0}^{\infty} x^m p_H(x) \tag{S7}$$

Once again, this method may be extended to multi-component systems.

Zero-information closure uses maximum-entropy moments as approximations for higher-order moments in simulation:

$$\underline{\mu}' = \underline{\mu}'_H \tag{S8}$$

Note that the order of closure, M, used throughout the section is applied without drawing any conclusions about accuracy. At present the order of closure is chosen in a trial-and-error fashion. In the future, error analysis may be used to determine the necessary or optimal closure order using ZI-Closure.

The method used within this study for the determination of the Lagrange parameters, $\underline{\lambda}$, given a set of known lower-order moments, $\underline{\mu}$, is a simple Newton-Raphson optimization scheme. The exact algorithm is provided below:

Algorithm S1: Newton-Raphson Method for Most-Likely Distribution Determination

1) Initial Lagrange parameters guess, $\underline{\lambda} = \underline{\lambda}_0$, is given. A range for each component is provided along with the values for the known moments that need to be matched, $\underline{\mu}$. Typically, the guess will be a set of zeros, although if a previous step in the simulation determined the parameters $\underline{\lambda}$ with reasonable accuracy those values can be used to speed up the simulation.

2) Calculate $p_H(x)$ using $\underline{\lambda}$ (Equation S6)

3) Calculate the lower-order maximum entropy moments, $\underline{\mu}_H$, using $p_H(x)$ (Equation S7)

4) Calculate the difference between the known moments and maximum entropy moments:

$$\Delta\underline{\mu} = \underline{\mu} - \underline{\mu}_H \tag{S9}$$

5) Calculate the 2-norm error:

$$\varepsilon = \Delta\underline{\mu}^T \Delta\underline{\mu} \tag{S10}$$

6) If ε ≤ tolerance proceed to (7). The tolerance was assumed to be the machine error ($1 \cdot 10^{-16}$) in this case. Else:

a. Calculate the Jacobian matrix J:

$$J_{i,j} = \frac{\partial \mu_{H,i}}{\partial \lambda_j} \tag{S11}$$

For a single component system the result is trivial (substituting Equation S6 into Equation S7) and is easily extended to multi-component systems:

$$J_{i,j} = \frac{\partial}{\partial \lambda_j}\left[\sum_{x=0}^{\infty} x^i \exp\left(-1 - \lambda_0 - \lambda_1 \mathrm{x} - \cdots - \lambda_{\mathrm{M}} \mathrm{x}^{\mathrm{M}}\right)\right] = -\langle x^{i+j} \rangle_H \tag{S12}$$

b. As with any Newton-Raphson system the method uses a first-order Taylor expansion:

$$\Delta \underline{\mu} \cong J \Delta \underline{\lambda} \tag{S13}$$

Thus an approximate parameter step is calculated:

$$\Delta \underline{\lambda} = J^{-1} \Delta \underline{\mu} \tag{S14}$$

c. This is used to get a new set of parameters:

$$\underline{\lambda} = \underline{\lambda} + \Delta \underline{\lambda} \tag{S15}$$

d. Return to (2)

7) Output $\underline{\lambda}$ for use in calculating the higher-order moments $\underline{\mu}'_{\mathrm{H}}$ using Equation (S7)

This basic Newton-Raphson scheme is used by the ODE time trajectory algorithm in determining the higher-order moments. This method is also modified to produce steady-state distributions for Algorithm S3 in Section S3.

Throughout the supplemental material the lower-order moments in the simulation are often described as being "known". Please note that while the values of the lower-order moments are not known through time *a priori,* at each time step these values will have been determined. This information is then utilized within the Newton-Raphson optimization scheme. The results of the ODE solving method will be, in fact, these lower order moments. The following example will serve to explain this further.

Example – Reversible Dimerization

The following is an example of how the above equations are calculated using a simple illustrative example of reversible dimerization:

$$2\,A \xrightarrow{k_1} B$$
$$B \xrightarrow{k_2} 2\,A$$

If we are looking to apply 2nd-order closure (approximate the 3rd-order moments and higher) then the moment vectors are:

$$\underline{\mu} = \begin{bmatrix} \langle A^0 \rangle \\ \langle A \rangle \\ \langle A^2 \rangle \end{bmatrix}, \text{ and } \underline{\mu}' = [\langle A^3 \rangle]$$

Note that since B can be calculated by knowing A and the initial amounts A_0 and B_0 there is only one independent component which we take to be A.

As for bounding the system the total number of monomer (A) in the system is $A_0 + 2\ B_0$. For each reaction two monomers are consumed or produced, so the sets $A \in \{0, 2, \dots, A_0 + 2\ B_0\}$ or $\{1, 3, \dots, A_0 + 2\ B_0\}$ represent the two possible component ranges for the system. At this point such considerations are important in ensuring the method operates correctly.

With the known lower-order moments, $\underline{\mu}$, the Lagrange parameters, $\underline{\lambda}$, can be determined. Since we know three moments there will be three parameters to determine:

$$\underline{\lambda} = \begin{bmatrix} \lambda_0 \\ \lambda_1 \\ \lambda_2 \end{bmatrix}$$

The final entropy maximized distribution takes the form:

$$p_H(x) = \exp\left(-1 - \lambda_0 - \lambda_1 x - \lambda_2 x^2\right)$$

This is determined using Algorithm S1. The most important part of this is the Jacobian which takes on a simple three by three form:

$$J = -\begin{bmatrix} \frac{\partial \langle A^0 \rangle_H}{\partial \lambda_0} & \frac{\partial \langle A^0 \rangle_H}{\partial \lambda_1} & \frac{\partial \langle A^0 \rangle_H}{\partial \lambda_2} \\ \frac{\partial \langle A \rangle_H}{\partial \lambda_0} & \frac{\partial \langle A \rangle_H}{\partial \lambda_1} & \frac{\partial \langle A \rangle_H}{\partial \lambda_2} \\ \frac{\partial \langle A^2 \rangle_H}{\partial \lambda_0} & \frac{\partial \langle A^2 \rangle_H}{\partial \lambda_1} & \frac{\partial \langle A^2 \rangle_H}{\partial \lambda_2} \end{bmatrix}$$

$$J = -\begin{bmatrix} \langle A^0 \rangle_H & \langle A \rangle_H & \langle A^2 \rangle_H \\ \langle A \rangle_H & \langle A^2 \rangle_H & \langle A^3 \rangle_H \\ \langle A^2 \rangle_H & \langle A \rangle_H & \langle A^4 \rangle_H \end{bmatrix}$$

Note the symmetry of the matrix and the necessity to determine much higher-order moments. This structure poses problems when the Jacobian is nearly singular (e.g. a delta functional) or when considering systems with nearly independent components.

This short example should provide enough information to understand and apply the ZI-closure method to a variety of systems. In Sections S2 and S3 this example will be further expanded to show how time trajectories and steady-state distributions can be determined.

Section S2 – Theoretical Foundation for the ODE Time Trajectory Method

This section is designed to explain how the moment equation time trajectories are solved for through time using ZI-closure. The section is split into an outline of the method, a short but thorough description of the ODE algorithm utilized, and closing with an illustrative example.

Start with an arbitrary reaction network with N components interacting according to R reactions. Note that we assume all reactions have elementary rate laws. The reaction network is fully described by a set of reaction rates, $\{k_1, k_2, \ldots, k_R\}$, and a stoichiometric matrix ν where $\nu_{i,j}$ corresponds to how component j is affected by reaction i.

The reaction network can then be transformed into a set of moment equations up to order M (for example $\langle X \rangle$ is a first-order moment, $\langle XY^2 \rangle$ is a third order moment, etc.) [Sotiropoulos 2009, Smadbeck 2012]:

$$\frac{\partial \underline{\mu}}{\partial t} = A\underline{\mu} + A'\underline{\mu}' \tag{S16}$$

The vector $\underline{\mu}$ is the set of lower-order moments of length $N_M = \binom{N+M}{M}$ and $\underline{\mu}'$ is the set of higher-order moments necessary for closure. If m additional orders of moments are needed for closure then $\underline{\mu}'$ is of size $N'_M = \binom{N+M+m-1}{M+m}$. This can be calculated using the N_M equation above. Typically, m is equal to one if second-order reactions are present, two if third-order reactions are present, etc. Using this notation A is an $N_M \times N_M$ matrix and A' is an $N_M \times N'_M$ matrix.

Using ZI-closure as described in Section S1, Equation S16 is closed by substituting in the maximum entropy higher-order moments (Equation S7):

$$\frac{\partial \underline{\mu}}{\partial t} = A\underline{\mu} + A'\underline{\mu}'_H \tag{S17}$$

By recalculating the Lagrange parameters, $\underline{\lambda}$, at each step any ODE solver can be used to determine time trajectories for the lower-order moments, $\underline{\mu}$. Herein we use the adaptive time step, 5th-order Runge-Kutta ODE solver (ode15s) built into MATLAB (The MathWorks Inc., Natick, MA).

Algorithm S2: ODE Time Trajectory Method

The following is a simple ODE solving method provided to illustrate how Algorithm S1 allows for the closure of Equation S16 and the determination of moment time trajectories.

1) Define $t = t_0$ and $\underline{\mu} = \underline{\mu}_0$. The moment matrices A and A' are known. The higher-order moments $\underline{\mu}' = \underline{\mu}'_0$ are initially known as well. A range for the system components is necessary for the Newton-Raphson optimization.

2) The first step is calculated with Equation S16 using the chosen ODE method obtaining $\Delta\mu$ and Δt.

3) Step forward: $\underline{\mu} = \underline{\mu} + \Delta\underline{\mu}$ and $t = t + \Delta t$. The higher-order moments $\underline{\mu}'$ are now unknown.

4) If $t \geq t_{max}$ proceed to (5), else:

    a. Use Algorithm S1 to obtain parameters $\underline{\lambda}$ from $\underline{\mu}$

    b. Use Equation S7 to obtain $\underline{\mu}'_H$ from $\underline{\lambda}$

    c. The next step, $\Delta\underline{\mu}$ and Δt, is calculated with Equation S17 with the chosen ODE method. Again, Δt is adaptive to account for system stiffness and particular to the chosen solver.

    d. Return to (3)

5) Return $\underline{t}$, a vector of desired time points, and the corresponding vector through time for each of the lower-order moments $\underline{\mu}$

The closure step is folded into the ODE solving scheme at step (4).

Example – Reversible Dimerization

See Section S1 for the beginning of this example problem. Here will be determined the appropriate A and A' matrices given the reversible dimerization reaction network:

$$2\,A \xrightarrow{k_1} B$$
$$B \xrightarrow{k_2} 2\,A$$

The method for determining A and A' is outlined in detail in a previous publication [Smadbeck 2012]. It can be noted that since $B = (A_0 + 2\,B_0 - A)/2$ there is only one independent component (N = 1). We are looking at 2nd-order closure (M = 2) and m is one because there is a second-order reaction. Therefore:

$$N_M = \binom{N+M}{M} = \frac{3!}{2!\,1!} = 3$$
$$N'_M = \binom{N+M}{M+1} = \frac{3!}{3!\,0!} = 1$$

Therefore A will be a 3 × 3 matrix, and A' will be a 3 × 1 matrix. Let $S_0 = A_0 + 2\,B_0$, and the two matrices can be determined to be:

$$A = \begin{bmatrix} 0 & 0 & 0 \\ k_2 S_0 & -k_2 & -k_1 \\ k_2 S_0 & 2k_2 S_0 - 3k_2 & -k_1 - 2k_2 \end{bmatrix} \text{ and } A' = \begin{bmatrix} 0 \\ 0 \\ -2k_1 \end{bmatrix}$$

At each step within the ODE solver A can be matrix multiplied by $\underline{\mu}$ and A' by $\underline{\mu}'$ to determine both $\Delta\underline{\mu}$ and by extension Δt.

The examples in Sections S1 and S2 now combine to show a clear picture of how the time trajectories using ZI-closure would be determined for a simple reversible dimerization system. This can be extended to a variety of different networks. In section S3 the example will conclude with an optimization method for immediately determining the steady-state distribution for the reversible dimerization network.

Section S3 – Theoretical Foundation for the Steady-State Determination Method

This section is designed to present the underlying theory for the one-step steady-state determination using ZI-closure. The section presents theory, the specific algorithm used to obtain steady-state distributions, and closes by completing the dimerization example from Sections S1 and S2.

Steady-state determination is structurally similar to Algorithm S1 with one key difference. In Section S1 lower-order moment values, $\underline{\mu}$, are known with given initial conditions (e.g. a Kronecker delta function or a Gaussian distribution). This information can then be used to obtain the desired information, the higher-order moments, $\underline{\mu}'_H$. In the case of steady-state determination the lower-order moments are instead the desired information, $\underline{\mu}_{SS}$, and must be obtained using other available information.

The information that is available, but was unused in Section S1, was the reaction network structure, Equation S16 from Section S2:

$$\frac{\partial \underline{\mu}}{\partial t} = A\underline{\mu} + A'\underline{\mu}'$$

In particular, take an augmented matrix B:

$$B = [A \mid A'] \tag{S18}$$

This matrix is N_M x (N_M+N_M') and is thus under-defined. The rank of B can be at most $N_M - 1$, depending on the reaction network. Given this idealized case the null space will have a rank of $N'_M + 1$. The null space basis can be described by an (N_M+N_M') x (N_M'+1) matrix with $N'_M + 1$ independent moments. The set of basis vectors for the null space can be manipulated such that it takes the form:

$$C_{SS} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ C_{1,1} & C_{1,2} & C_{1,3} & \cdots & C_{1,N'_M+1} \\ C_{2,1} & C_{2,2} & C_{2,3} & \cdots & C_{2,N'_M+1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ C_{N_M,1} & C_{N_M,2} & C_{N_M,3} & \cdots & C_{N_M,N'_M+1} \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \tag{S19}$$

The entry C(m,n) indicates how the m^{th}-moment scales with the n^{th} independent moment with respect to the null-space of matrix B. The independent moments can, for the most part, be chosen arbitrarily. We typically choose the independent moments to be the higher-order moments and the 0^{th} moment (which is always independent). Any linear combination of the columns in C_{SS} (i.e. C_{SS} matrix multiplied by any vector) is in the null space of the matrix B.

This basis set is the additional information needed to obtain steady-states: a relationship between the lower-order moments and higher-order moments has now been established that must be satisfied by the steady-state distribution. This relationship results in a modified objective function:

$$\begin{bmatrix}\Delta\underline{\mu}_{SS}\\ \Delta\underline{\mu}'_{SS}\end{bmatrix} = \begin{bmatrix}\underline{\mu}_H\\ \underline{\mu}'_H\end{bmatrix} - \begin{bmatrix}\underline{\mu}_{SS}\\ \underline{\mu}'_{SS}\end{bmatrix} = \begin{bmatrix}\underline{\mu}_H\\ \underline{\mu}'_H\end{bmatrix} - C_{SS}\begin{bmatrix}1\\ \underline{\mu}'_H\end{bmatrix} \quad \text{(S20)}$$

Note that the final form takes into account our choice of independent moments as the 0^{th} moment (always equal to one) and the higher order moments, $\underline{\mu}'_H$. Using the modified objective function, $\Delta\underline{\mu}_{SS}$ in place of $\Delta\underline{\mu}$, the same Newton-Raphson algorithm from Section S1 can be used to obtain steady-state distributions.

When generating the results for this manuscript an expanded Jacobian was used instead. The expanded Jacobian matrix is an $(N_M + N_M')$x$(N_M + N_M')$ matrix whereas in Section S1 it was an N_M x N_M matrix. The higher-order moments, μ', are now present as both matched variables and fitted parameters. We have found that using an expanded Jacobian improves convergence to the steady-state solution.

Algorithm S3: Steady-State Newton-Raphson Method

1) Initial parameter guess, $\underline{\lambda}_{SS}=\underline{\lambda}_{SS,0}$, is given. A range for the components is also provided. Typically a guess is a vector of zeros, but if a previous step gave a reasonably close distribution the output parameters can be utilized as an initial guess.
2) Calculate the null space (the matrix C_{SS}) for Equation S18.
3) Calculate $p_H(x)$ using $\underline{\lambda}_{SS}$ (Equation S6)
4) Calculate $\underline{\mu}_H$ and $\underline{\mu}'_H$ using $p_H(x)$ (Equation S7).
5) The calculated moments, $\underline{\mu}_H$ and $\underline{\mu}'_H$, are then compared to those given by the null space (Equation S19 and S20):
$$\begin{bmatrix}\Delta\underline{\mu}_{SS}\\ \Delta\underline{\mu}'_{SS}\end{bmatrix} = \begin{bmatrix}\underline{\mu}_H\\ \underline{\mu}'_H\end{bmatrix} - \begin{bmatrix}\underline{\mu}_{SS}\\ \underline{\mu}'_{SS}\end{bmatrix} = \begin{bmatrix}\underline{\mu}_H\\ \underline{\mu}'_H\end{bmatrix} - C_{SS}\begin{bmatrix}1\\ \underline{\mu}'_H\end{bmatrix}$$
6) Calculate the 2-norm error:
$$\varepsilon_{SS} = \begin{bmatrix}\Delta\underline{\mu}_{SS}\\ \Delta\underline{\mu}'_{SS}\end{bmatrix}^T \begin{bmatrix}\Delta\underline{\mu}_{SS}\\ \Delta\underline{\mu}'_{SS}\end{bmatrix} \quad \text{(S21)}$$
7) If ε_{SS} < tolerance proceed to (8). Note again that the tolerance was taken to be $1\cdot10^{-16}$, the machine tolerance, but with a catch to prevent infinite loops. Else:
    a. Calculate the expanded Jacobian, J_{SS}. This is similar to the Jacobian in Algorithm S1, but expanded to include the higher order moments, $\underline{\mu}'_H$, as rows and the corresponding null space basis vectors as new columns. Essentially, the information for determining the steady-state lower-order moments are provided by the null space. The Jacobian is an $(N_M + N_M')$x$(N_M + N_M')$ matrix (see section S2) and takes the form:
$$J_{SS} = \left[\begin{array}{c|c}\dfrac{\partial\underline{\mu}_H}{\partial\underline{\lambda}} & \\ \dfrac{\partial\underline{\mu}'_H}{\partial\underline{\lambda}} & C'_{SS}\end{array}\right] \quad \text{(S22)}$$
    Here C'_{SS} denotes the null space basis vectors corresponding to just the higher-order moments. The Jacobian from Algorithm S1 is present as the top-left corner of the

steady-state Jacobian (Equation S11 and S12). See the example at the end of this section for a clearer view on how this steady-state Jacobian is developed.

b. As with any Newton-Raphson scheme the method uses a first order Taylor expansion:

$$\begin{bmatrix}\Delta\underline{\mu}_{SS}\\ \Delta\underline{\mu}'_{SS}\end{bmatrix} \cong J_{SS}\begin{bmatrix}\Delta\underline{\lambda}_{SS}\\ \Delta\underline{\mu}'_{SS}\end{bmatrix} \tag{S23}$$

Note that the higher-order moments are used as both a variable and a parameter in the steady-state algorithm. In reality you are not fitting the higher-order moments, but rather they are providing the null-space information.

c. An approximate parameter step is calculated:

$$\begin{bmatrix}\Delta\underline{\lambda}_{SS}\\ \Delta\underline{\mu}'_{SS}\end{bmatrix} = J_{SS}^{-1}\begin{bmatrix}\Delta\underline{\mu}_{SS}\\ \Delta\underline{\mu}'_{SS}\end{bmatrix} \tag{S24}$$

d. A new parameter set is determined:

$$\underline{\lambda}_{SS} = \underline{\lambda}_{SS} + \Delta\underline{\lambda}_{SS} \tag{S25}$$

e. Return to (3)

8) Output $\underline{\lambda}_{SS}$ for use in calculating $\underline{\mu}_{SS}$ using Equation S7.

The three algorithms presented in Sections S1-3 represent a method for the stochastic simulation of chemical reaction networks that is totally independent from Gillespie's stochastic simulation algorithm and its descendents. The steady-state algorithm presented within this section in particular marks, potentially, the first universal one-step method for stochastic steady-state distribution determination.

Example – Reversible Dimerization

This section extends the examples in Section S1 and S2. Recall that the reaction network being explored is a simple reversible dimerization network:

$$2\,A \xrightarrow{k_1} B$$
$$B \xrightarrow{k_2} 2\,A$$

For the 2nd-order ZI-Closure recall that the lower and higher-order moment vectors are:

$$\underline{\mu} = \begin{bmatrix}\langle A^0\rangle\\ \langle A\rangle\\ \langle A^2\rangle\end{bmatrix}, \text{ and } \underline{\mu}' = [\langle A^3\rangle]$$

The matrices in Equation S16 are thus:

$$A = \begin{bmatrix}0 & 0 & 0\\ k_2S_0 & -k_2 & -k_1\\ k_2S_0 & 2k_2S_0 - 3k_2 & -k_1 - 2k_2\end{bmatrix} \text{ and } A' = \begin{bmatrix}0\\ 0\\ -2k_1\end{bmatrix}$$

These can be used to create the augmented matrix B in Equation S18:

$$B = \begin{bmatrix}0 & 0 & 0 & 0\\ k_2S_0 & -k_2 & -k_1 & 0\\ k_2S_0 & 2k_2S_0 - 3k_2 & -k_1 - 2k_2 & -2k_1\end{bmatrix}$$

The null space is thus simple to obtain:

$$C_{SS} = \begin{bmatrix} 1 & 0 \\ \dfrac{S_0 k_2}{k_2 + k_1(S_0 - 1)} & \dfrac{k_1^2}{k_2\left(k_2 + k_1(S_0 - 1)\right)} \\ \dfrac{S_0 k_2 (S_0 - 1)}{k_2 + k_1(S_0 - 1)} & \dfrac{-k_1}{k_2 + k_1(S_0 - 1)} \\ 0 & 1 \end{bmatrix}$$

C_{ss} has two columns, one corresponding to the 0th moment (necessarily independent since it is invariant), and one corresponding to the single higher-order moment necessary for closure, $\langle A^3 \rangle$.

The null basis vector corresponding to $\langle A^3 \rangle$ is incorporated into the expanded Jacobian by including $\langle A^3 \rangle$ as both a matched variable and parameters. So the variables to be matched are: $\{\langle A^0 \rangle_{SS}, \langle A \rangle_{SS}, \langle A^2 \rangle_{SS}, \langle A^3 \rangle_{SS}\}$, and the parameters to be fit are: $\{\lambda_0, \lambda_1, \lambda_2, \langle A^3 \rangle_{SS}\}$:

$$J_{SS} = \begin{bmatrix} \dfrac{\partial \langle A^0 \rangle_{SS}}{\partial \lambda_0} & \dfrac{\partial \langle A^0 \rangle_{SS}}{\partial \lambda_1} & \dfrac{\partial \langle A^0 \rangle_{SS}}{\partial \lambda_2} & \dfrac{\partial \langle A^0 \rangle_{SS}}{\partial \langle A^3 \rangle_{SS}} \\ \dfrac{\partial \langle A \rangle_{SS}}{\partial \lambda_0} & \dfrac{\partial \langle A \rangle_{SS}}{\partial \lambda_1} & \dfrac{\partial \langle A \rangle_{SS}}{\partial \lambda_2} & \dfrac{\partial \langle A \rangle_{SS}}{\partial \langle A^3 \rangle_{SS}} \\ \dfrac{\partial \langle A^2 \rangle_{SS}}{\partial \lambda_0} & \dfrac{\partial \langle A^2 \rangle_{SS}}{\partial \lambda_1} & \dfrac{\partial \langle A^2 \rangle_{SS}}{\partial \lambda_2} & \dfrac{\partial \langle A^2 \rangle_{SS}}{\partial \langle A^3 \rangle_{SS}} \\ \dfrac{\partial \langle A^3 \rangle_{SS}}{\partial \lambda_0} & \dfrac{\partial \langle A^3 \rangle_{SS}}{\partial \lambda_1} & \dfrac{\partial \langle A^3 \rangle_{SS}}{\partial \lambda_2} & \dfrac{\partial \langle A^3 \rangle_{SS}}{\partial \langle A^3 \rangle_{SS}} \end{bmatrix}$$

Note that because C_{SS} describes a relationship between the lower-order moments and higher-order moments it can be used for the last column in the Jacobian. The rest are found in the same manner as Equation S12:

$$J_{SS} = \begin{bmatrix} -\langle A^0 \rangle & -\langle A \rangle & -\langle A^2 \rangle & 0 \\ -\langle A \rangle & -\langle A^2 \rangle & -\langle A^3 \rangle & \dfrac{k_1^2}{k_2\left(k_2 + k_1(S_0 - 1)\right)} \\ -\langle A^2 \rangle & -\langle A^3 \rangle & -\langle A^4 \rangle & \dfrac{-k_1}{k_2 + k_1(S_0 - 1)} \\ -\langle A^3 \rangle & -\langle A^4 \rangle & -\langle A^5 \rangle & 1 \end{bmatrix}$$

Section S4 – Additional Data and Figures

This section presents more results collected for models 1-3 from the body of the work. These additional results further substantiate the conclusions drawn in the main manuscript body. Furthermore, many of the lower-order ZI-Closure results are shown to provide insight on how increasing the order of ZI-Closure affects accuracy.

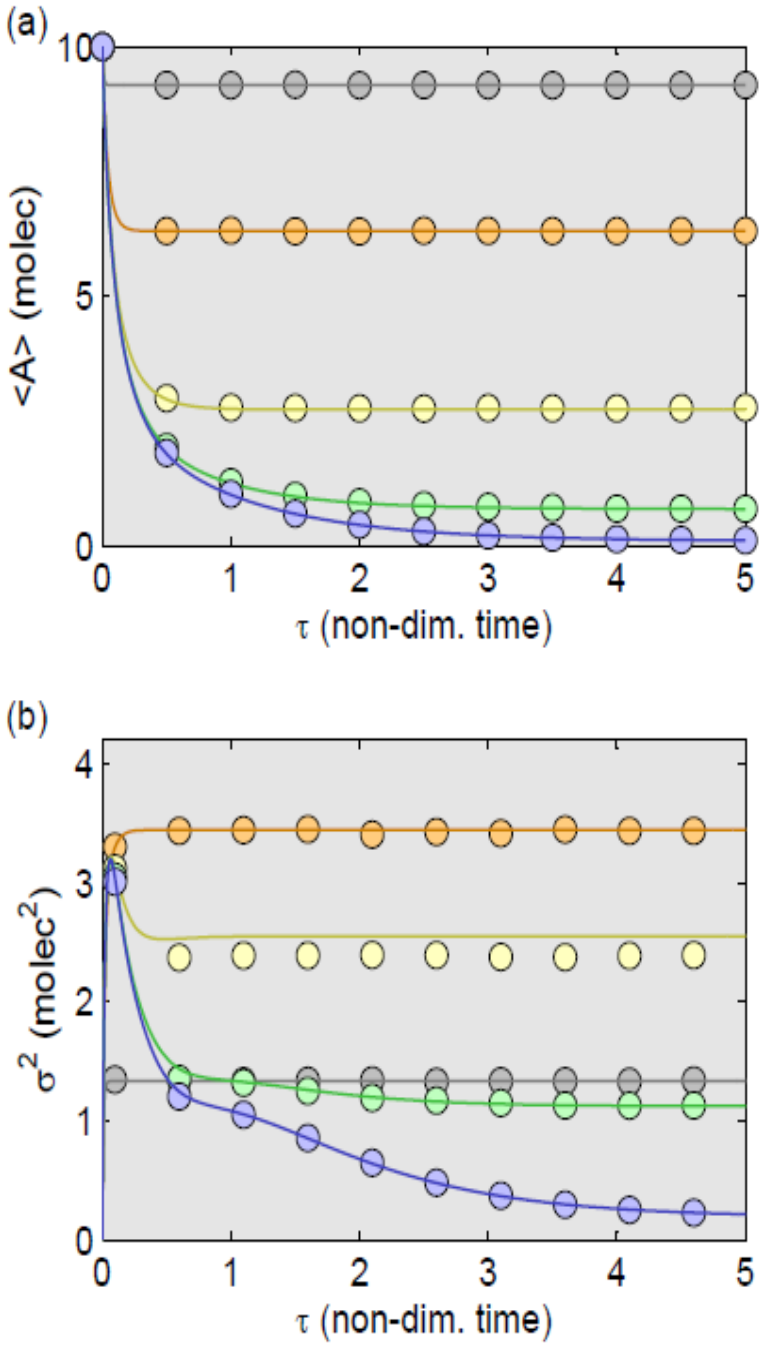

Figure S1.1 – Reversible Dimerization, Time Trajectory Results with 2nd-Order ZI-Closure

Evolution of the (a) average and (b) variance of the number of A molecules using 2nd-order ZI-closure (solid lines). Different colored circles represent SSA results (100000 trajectories) for various values of equilibrium constant K= k_2 / k_1. The gray (K = 100), orange (K = 10), yellow (K = 1), green (K = 0.1), and blue (K = 0.01) circles range across 5 orders of magnitude to demonstrate universailty of the method. Note that with second order closure the ZI-closure results differ substantially from the SSA when K is near one. The initial conditions are provided in the body of the manuscript.

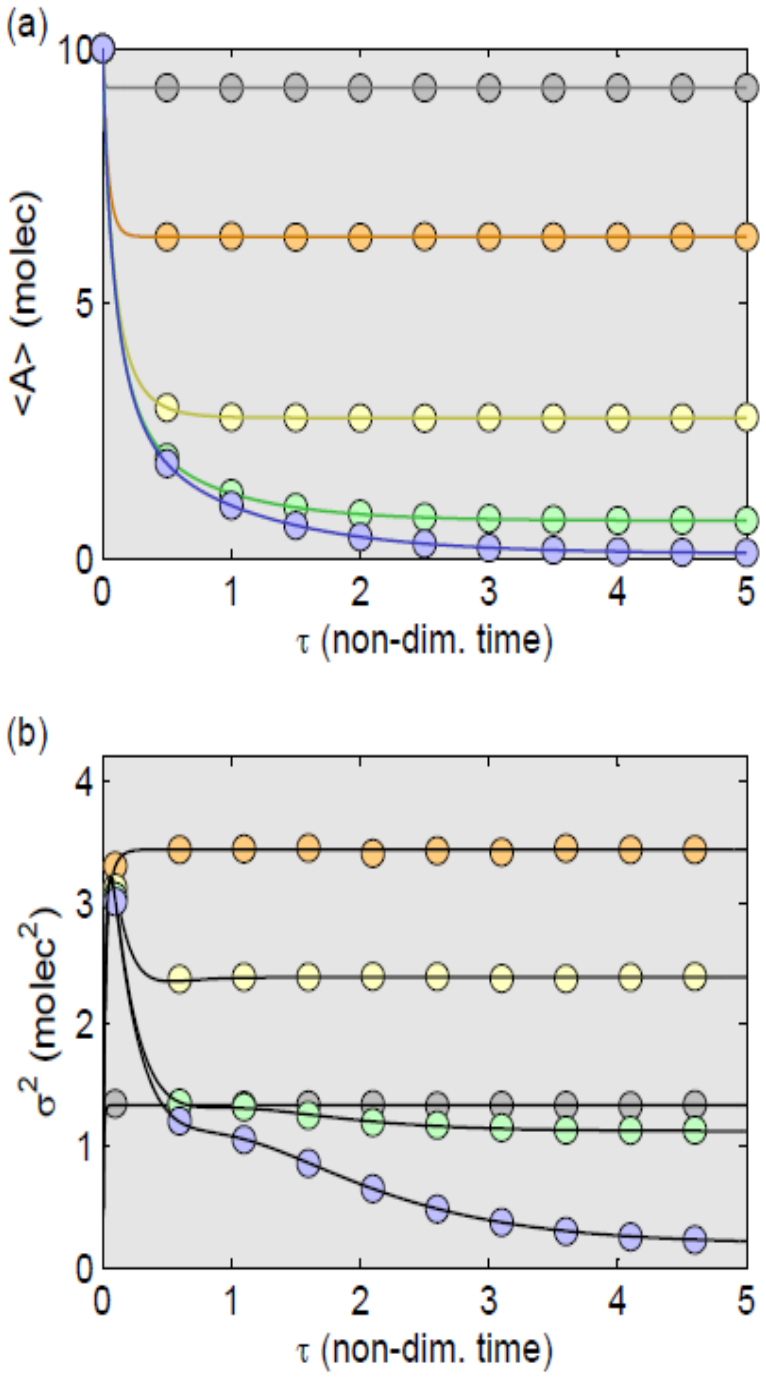

Figure S1.2 – Reversible Dimerization, Time Trajectory Results with 4th-Order ZI-Closure

Evolution of the (a) average and (b) variance of the number of A molecules using 4th-order ZI-closure (solid lines). Different colored circles represent SSA results (100000 trajectories) for various values of equilibrium constant K= k_2 / k_1. The gray (K = 100), orange (K = 10), yellow (K = 1), green (K = 0.1), and blue (K = 0.01) circles range across 5 orders of magnitude to demonstrate universailty of the method. The initial conditions are provided in the body of the manuscript.

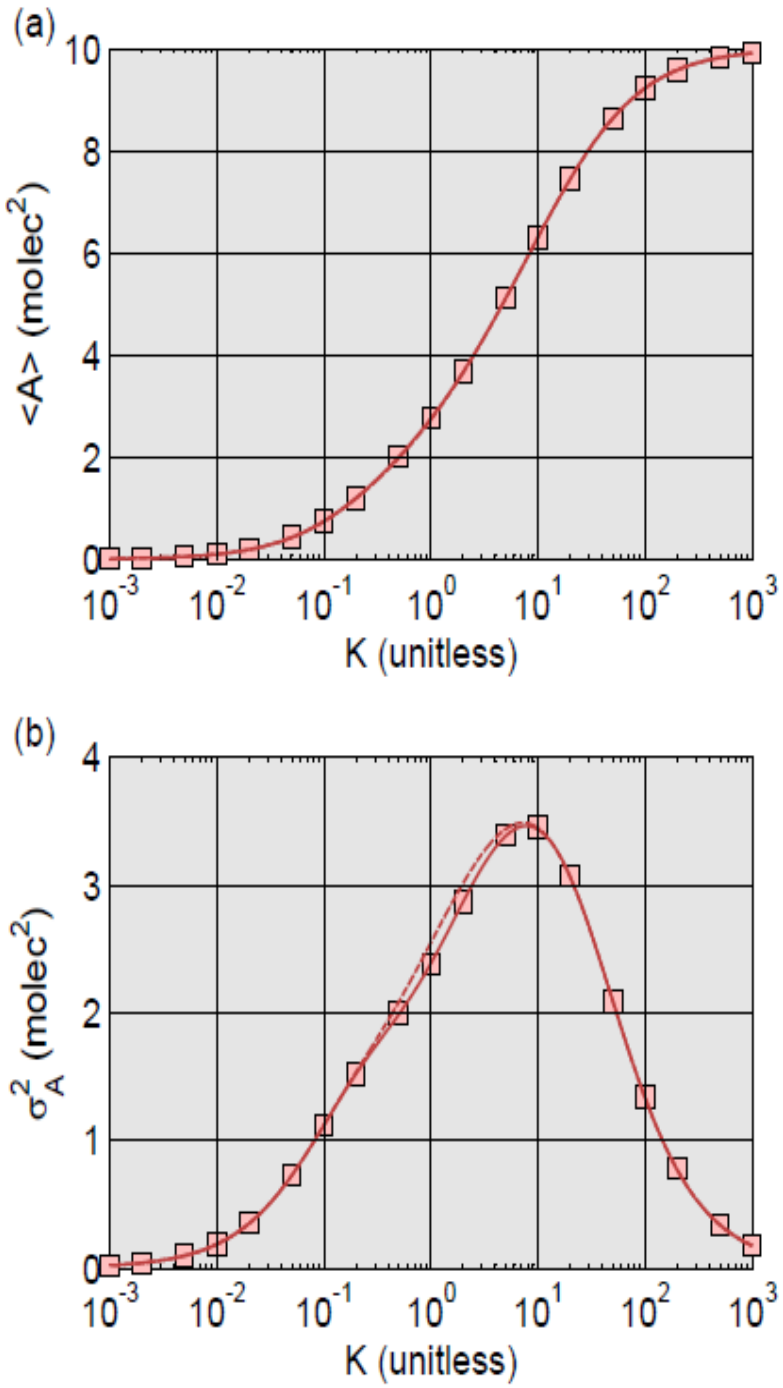

Figure S2 – Reversible Dimerization, Steady-State Results

The (a) average and (b) variance of the steady-state number of A molecules using 2nd (dashed lines) and 4th-order ZI-closure (solid lines). The results range across seven orders of magnitude for the equilibrium constant K= k_2 / k_1 (K = 10^{-3} to 10^3). A comparison is made to the SSA results (squares) using 100000 trajectories. Note the difference in accuracy near K equal to one for the two closure orders.

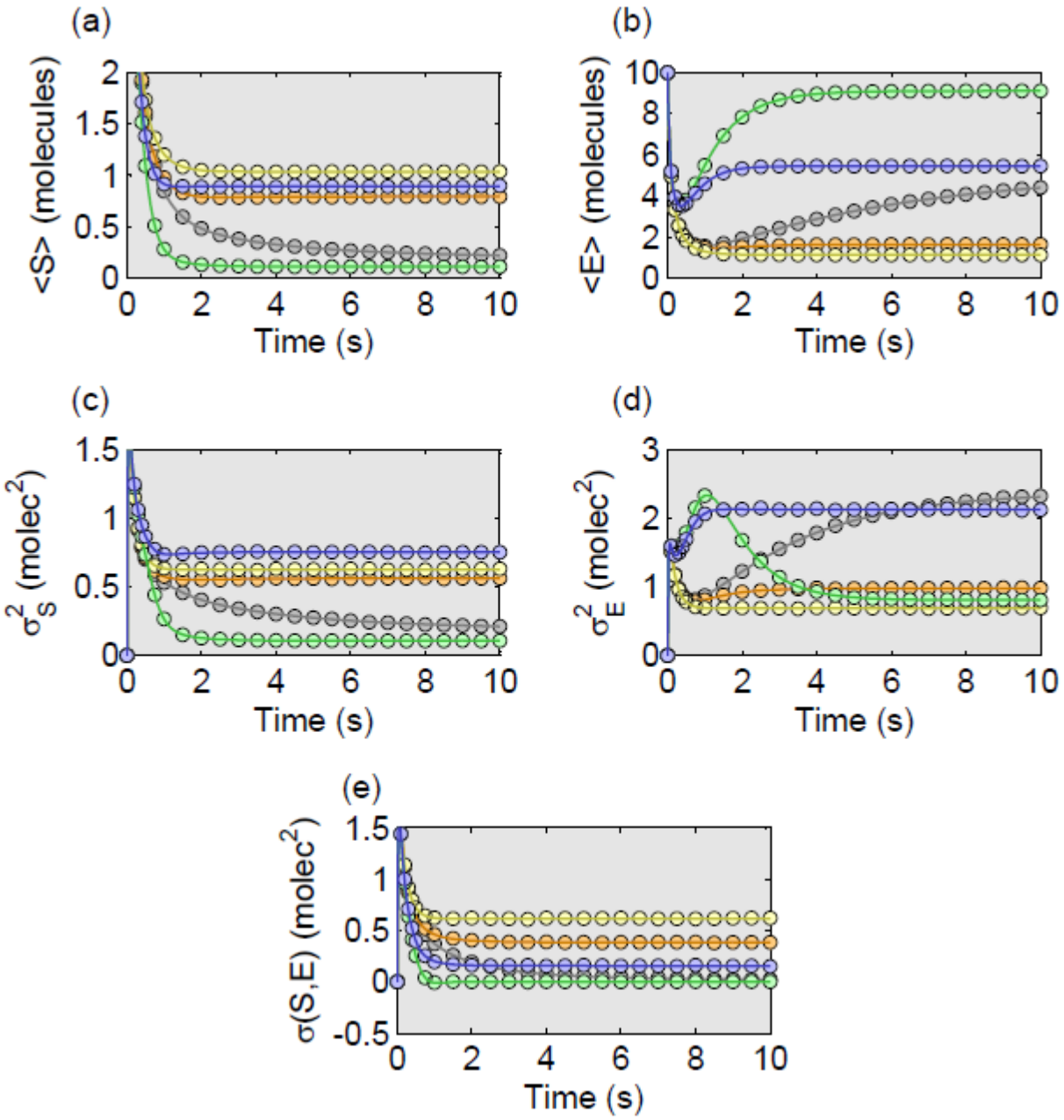

Figure S3.1 – Michaelis-Menten Time Trajectories for Parameters Sets 1-5

The (a) mean S, (b) mean E, (c) variance for S, (d) variance for E, and (e) covariance for the time trajectories for five parameters sets are shown using 4th-order ZI-closure (solid lines). The results are compared to SSA results ($1 \cdot 10^6$ trajectories, circles, colors indicated below). The parameter sets are: [k_1, k_2, k_3, k_4] = [1, 0.1, 0.1, 0.1] (gray), [1, 0.1, 0.1, 1] (orange), [1, 0.1, 0.1, 10] (yellow), [1, 0.1, 1, 0.1] (green), [1, 0.1, 1, 1] (blue). The initial conditions are provided in the body of the manuscript.

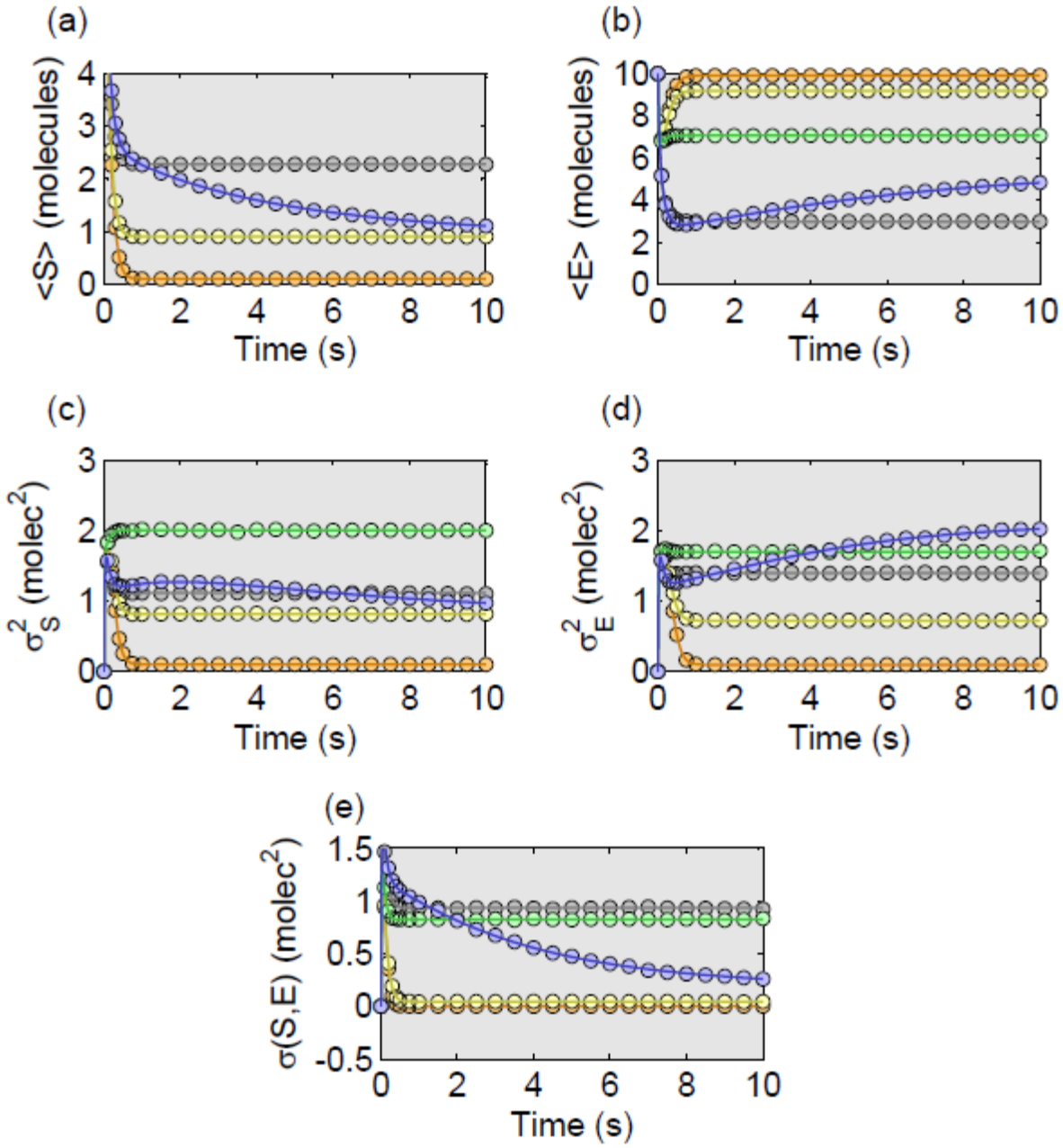

Figure S3.2 – Michaelis-Menten Time Trajectories for Parameters Sets 6-10

The (a) mean S, (b) mean E, (c) variance for S, (d) variance for E, and (e) covariance for the time trajectories for five parameters sets are shown using 4th-order ZI-closure (solid lines). The results are compared to SSA results ($1 \cdot 10^6$ trajectories, circles, colors indicated below). The parameter sets are: [k_1, k_2, k_3, k_4] = [1, 0.1, 1, 10] (gray), [1, 0.1, 10, 0.1] (orange), [1, 0.1, 10, 1] (yellow), [1, 0.1, 10, 10] (green), [1, 1, 0.1, 0.1] (blue). The initial conditions are provided in the body of the manuscript.

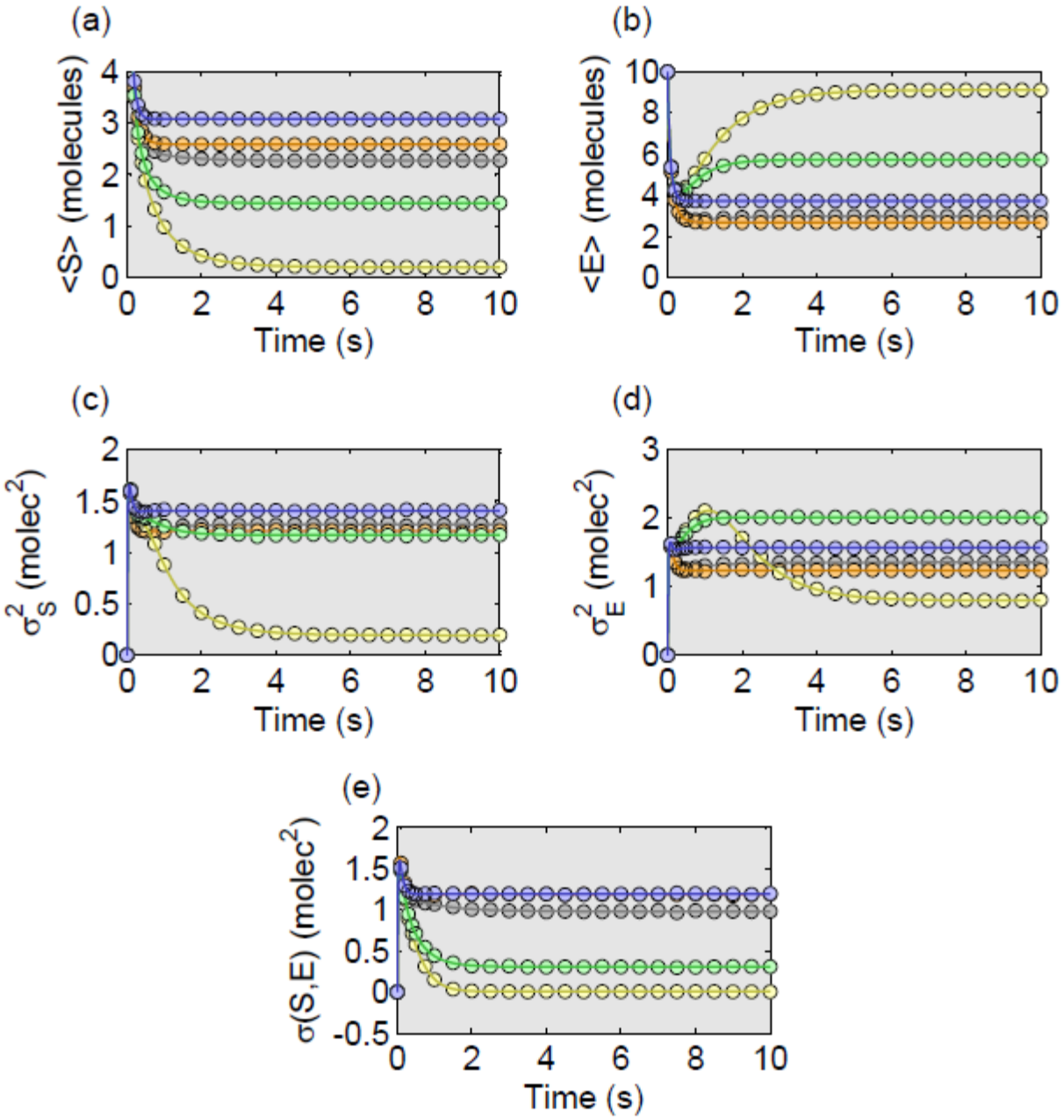

Figure S3.3 – Michaelis-Menten Time Trajectories for Parameters Sets 11-15

The (a) mean S, (b) mean E, (c) variance for S, (d) variance for E, and (e) covariance for the time trajectories for five parameters sets are shown using 4th-order ZI-closure (solid lines). The results are compared to SSA results ($1 \cdot 10^6$ trajectories, circles, colors indicated below). The parameter sets are: [k_1, k_2, k_3, k_4] = [1, 1, 0.1, 1] (gray), [1, 1, 0.1, 10] (orange), [1, 1, 1, 0.1] (yellow), [1, 1, 1, 1] (green), [1, 1, 1, 10] (blue). The initial conditions are provided in the body of the manuscript.

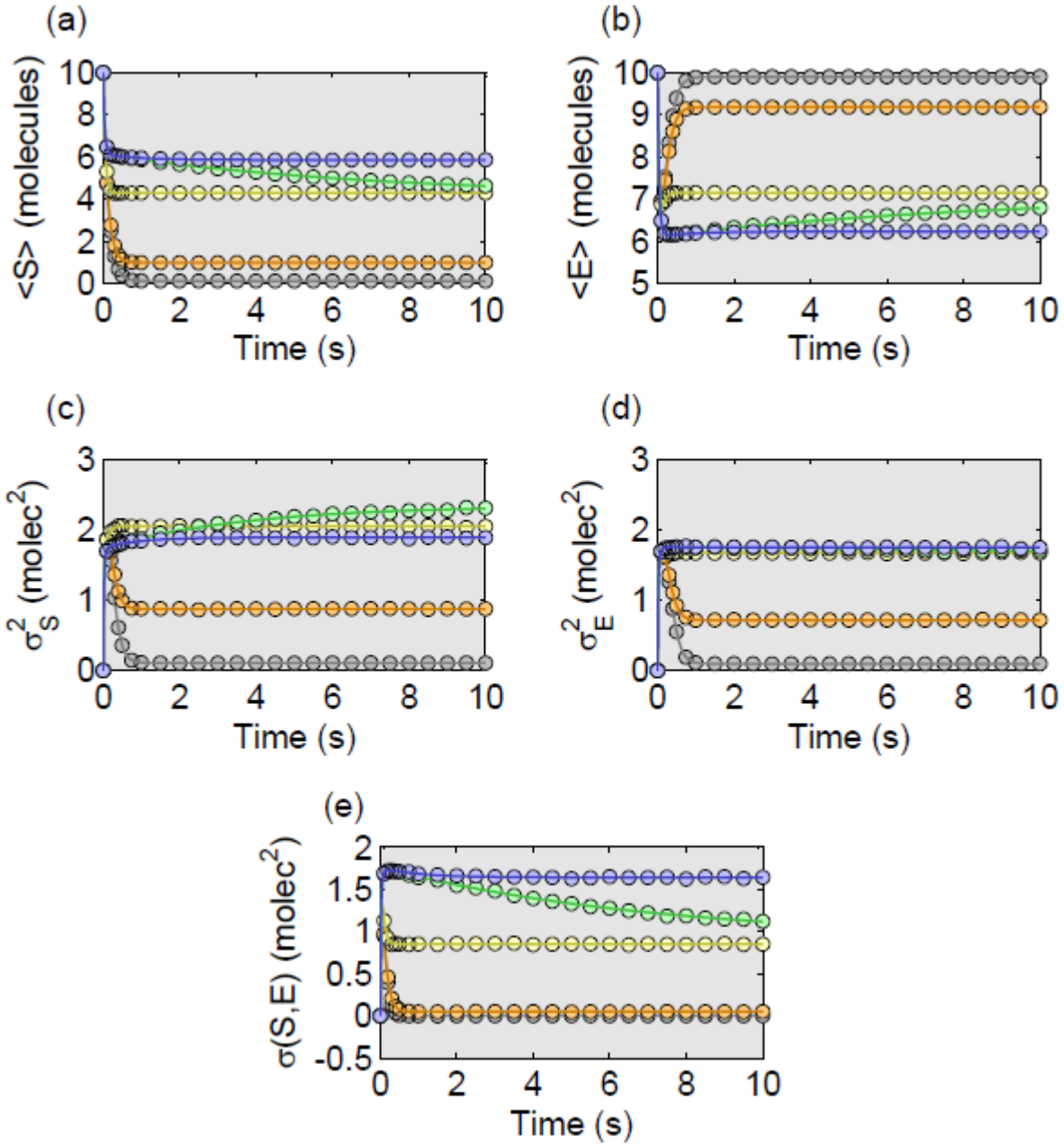

Figure S3.4 – Michaelis-Menten Time Trajectories for Parameters Sets 16-20

The (a) mean S, (b) mean E, (c) variance for S, (d) variance for E, and (e) covariance for the time trajectories for five parameters sets are shown using 4th-order ZI-closure (solid lines). The results are compared to SSA results ($1 \cdot 10^6$ trajectories, circles, colors indicated below). The parameter sets are: [k_1, k_2, k_3, k_4] = [1, 1, 10, 0.1] (gray), [1, 1, 10, 1] (orange), [1, 1, 10, 10] (yellow), [1, 10, 0.1, 0.1] (green), [1, 10, 0.1, 1] (blue). The initial conditions are provided in the body of the manuscript.

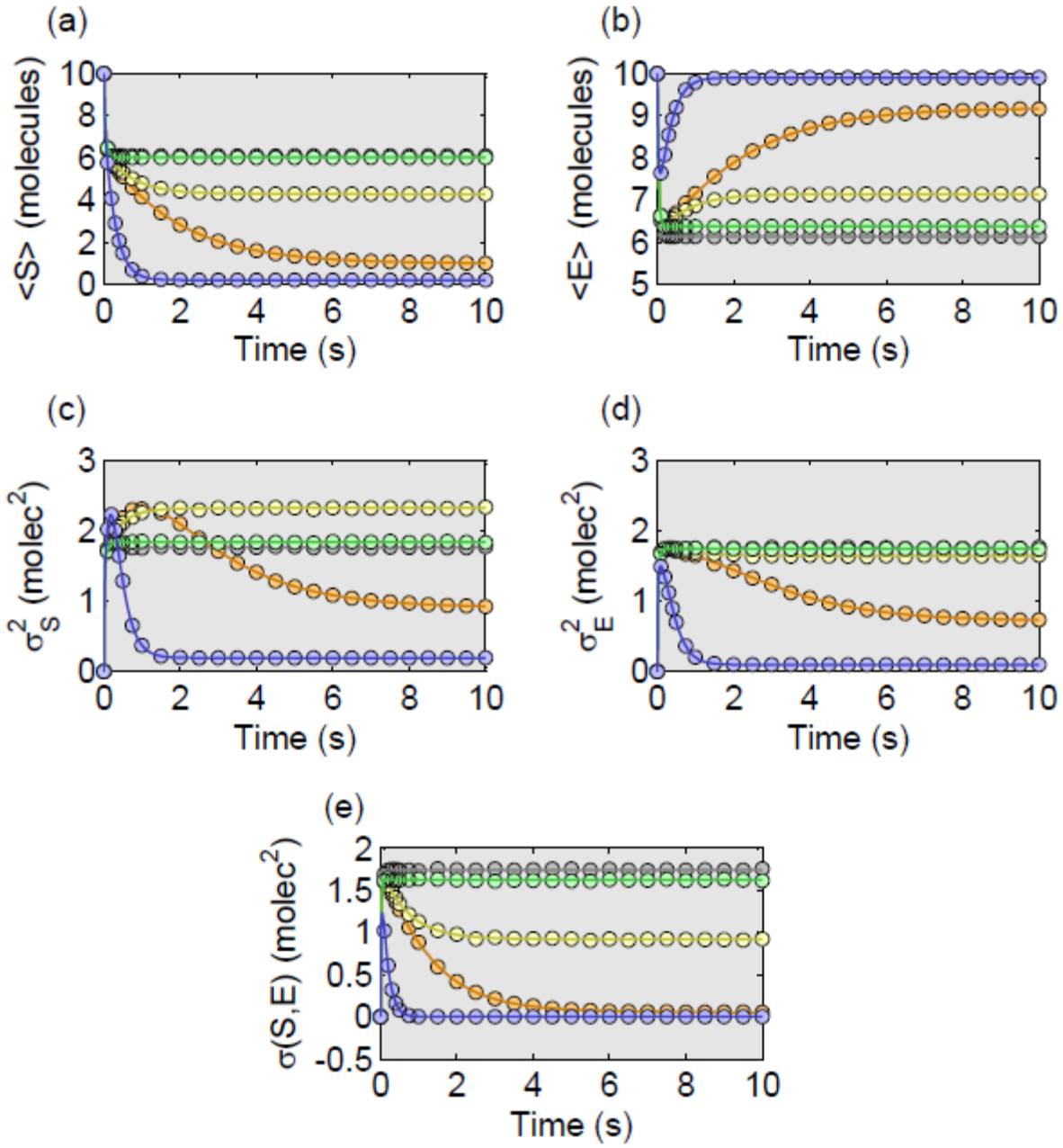

Figure S3.5 – Michaelis-Menten Time Trajectories for Parameters Sets 21-25

The (a) mean S, (b) mean E, (c) variance for S, (d) variance for E, and (e) covariance for the time trajectories for five parameters sets are shown using 4th-order ZI-closure (solid lines). The results are compared to SSA results ($1 \cdot 10^6$ trajectories, circles, colors indicated below). The parameter sets are: [k_1, k_2, k_3, k_4] = [1, 10, 0.1, 10] (gray), [1, 10, 1, 0.1] (orange), [1, 10, 1, 1] (yellow), [1, 10, 1, 10] (green), [1, 10, 10, 0.1] (blue). The initial conditions are provided in the body of the manuscript.

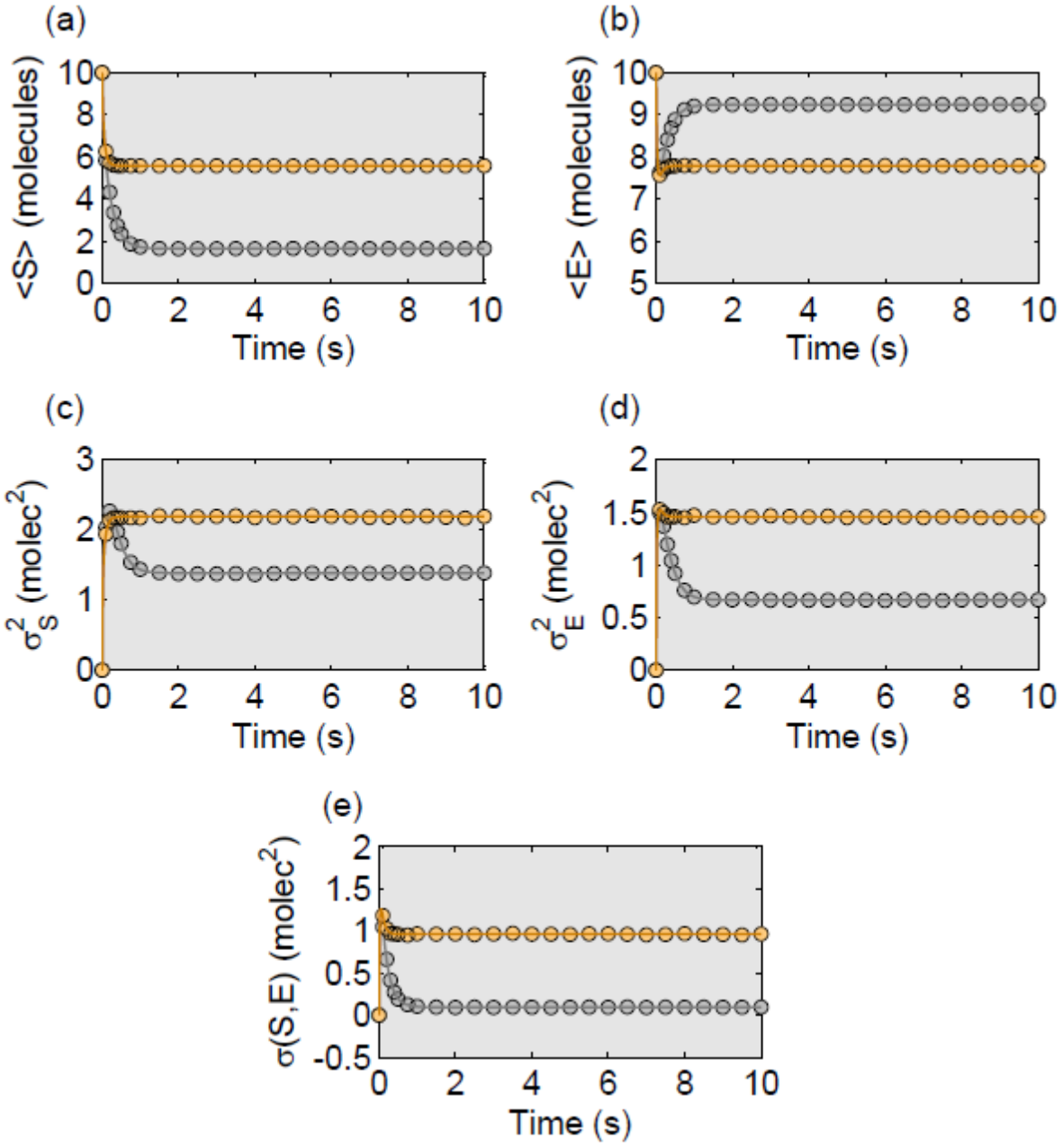

Figure S3.6 – Michaelis-Menten Time Trajectories for Parameters Sets 26 and 27

The (a) mean S, (b) mean E, (c) variance for S, (d) variance for E, and (e) covariance for the time trajectories for five parameters sets are shown using 4th-order ZI-closure (solid lines). The results are compared to SSA results ($1 \cdot 10^6$ trajectories, circles, colors indicated below). The parameter sets are: [k_1, k_2, k_3, k_4] = [1, 10, 10, 1] (gray), [1, 10, 10, 10] (orange). The initial conditions are provided in the body of the manuscript.

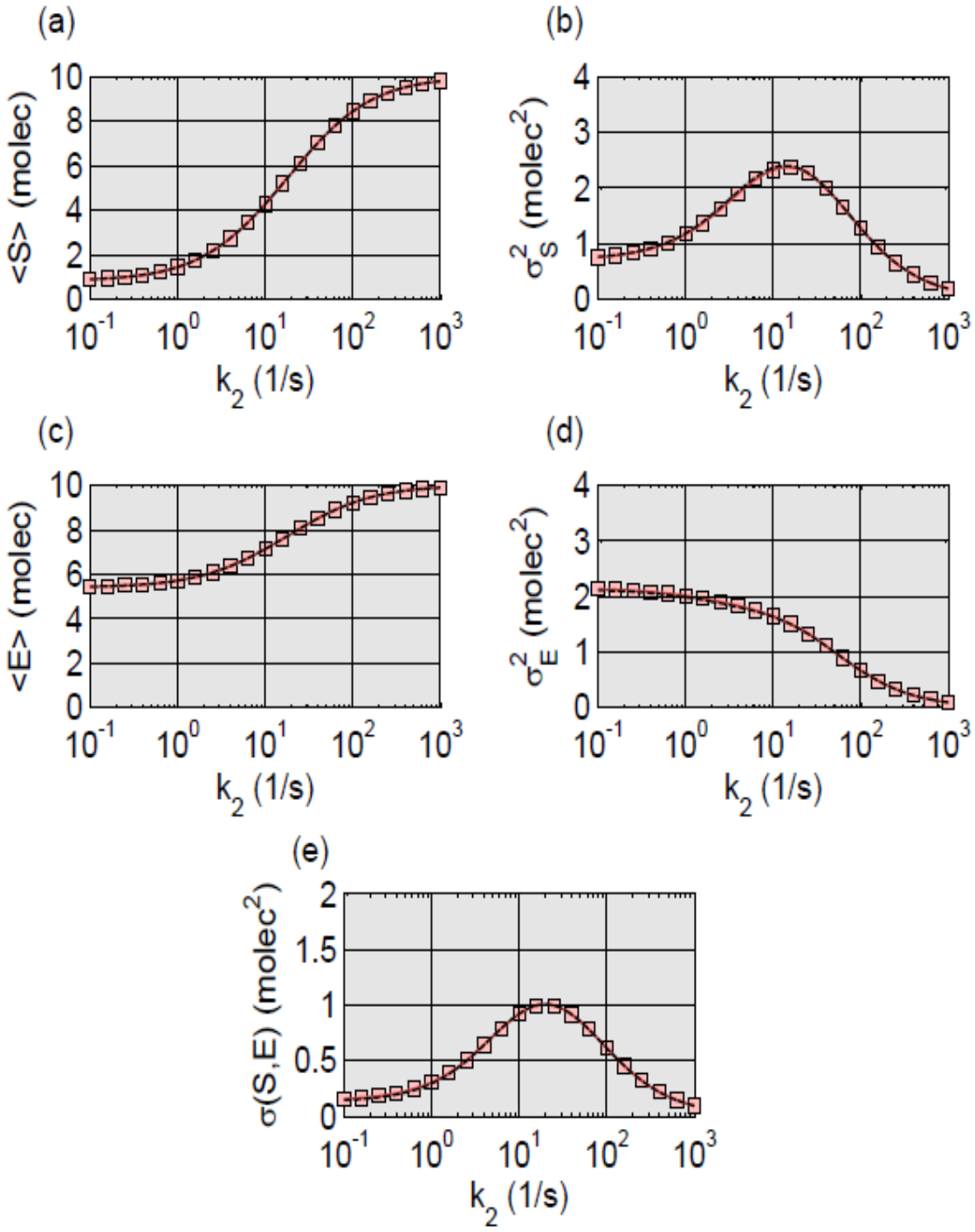

Figure S4.1 – Michaelis-Menten Steady-State Results across a range for k_1

The (a) mean S, (b) mean E, (c) variance for S, (d) variance for E, and (e) covariance for the steady-state Michaelis-Menten model with 2nd-order (dashed line) and 4th-order ZI-closure (solid line) compared to SSA results ($1 \cdot 10^6$ trajectories, squares). The value for k_1 is changed and all other parameters are set to one. Note that here there is no noticeable difference between 2nd and 4th-order closure.

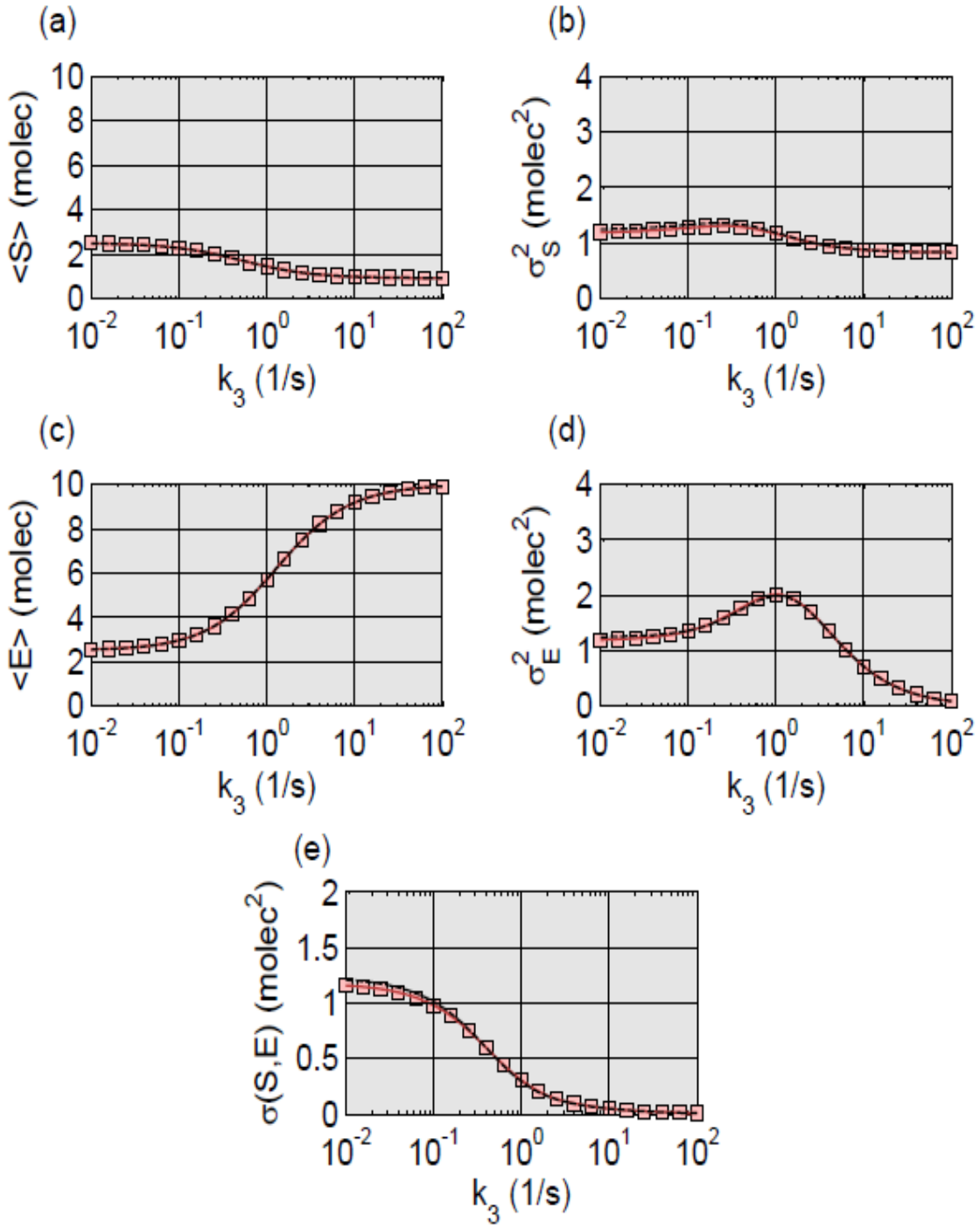

Figure S4.2 – Michaelis-Menten Steady-State Results across a range for k_2

The (a) mean S, (b) mean E, (c) variance for S, (d) variance for E, and (e) covariance for the steady-state Michaelis-Menten model with 2nd-order (dashed line) and 4th-order ZI-closure (solid line) compared to SSA results ($1 \cdot 10^6$ trajectories, squares). The value for k_1 is changed and all other parameters are set to one. Note that here there is no noticeable difference between 2nd and 4th-order closure.

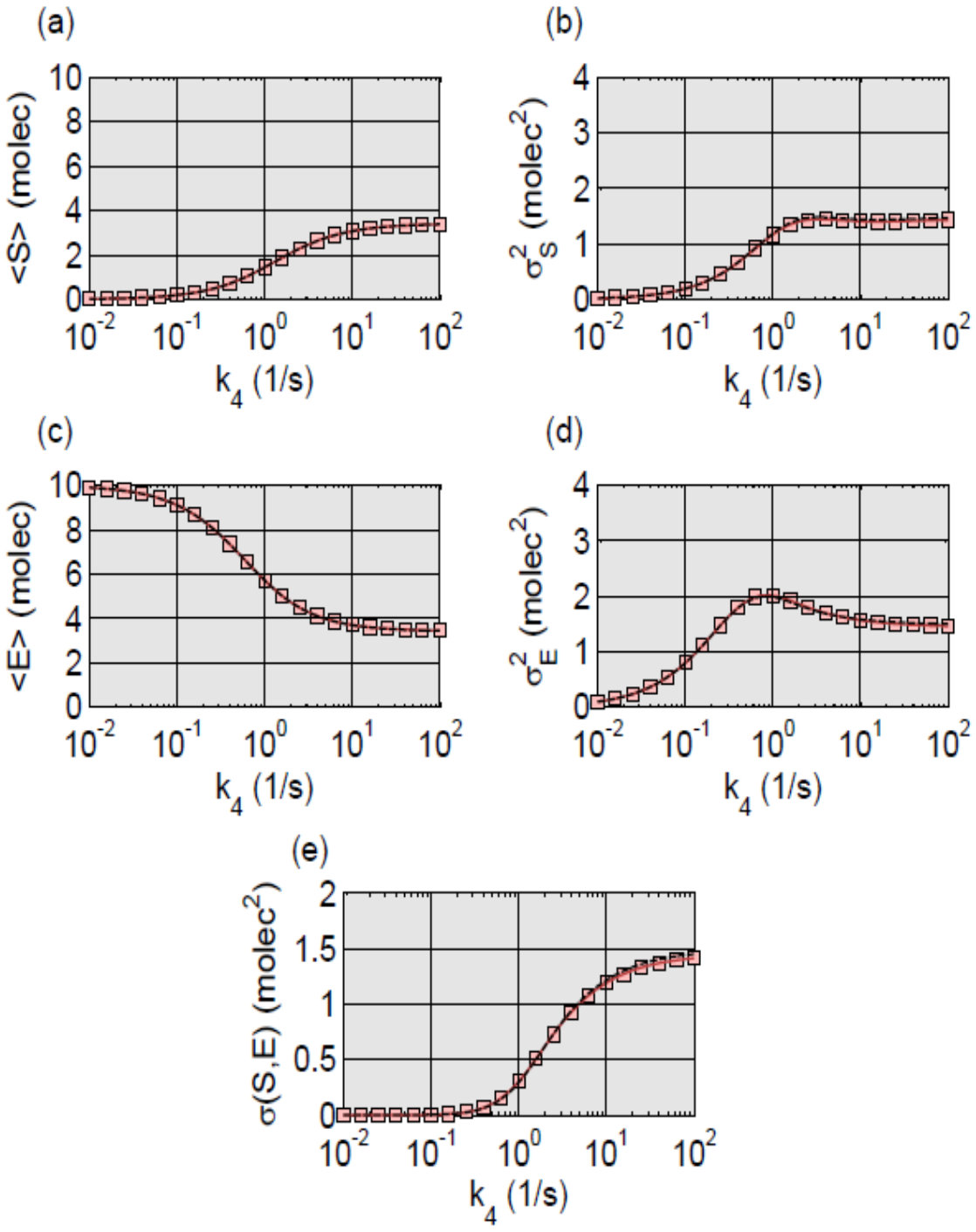

Figure S4.3 – Michaelis-Menten Steady-State Results across a range for k_3

The (a) mean S, (b) mean E, (c) variance for S, (d) variance for E, and (e) covariance for the steady-state Michaelis-Menten model with 2nd-order (dashed line) and 4th-order ZI-closure (solid line) compared to SSA results ($1 \cdot 10^6$ trajectories, squares). The value for k_1 is changed and all other parameters are set to one. The difference between 2nd and 4th-order closure is evident in the three 2nd-order plots (c-e).

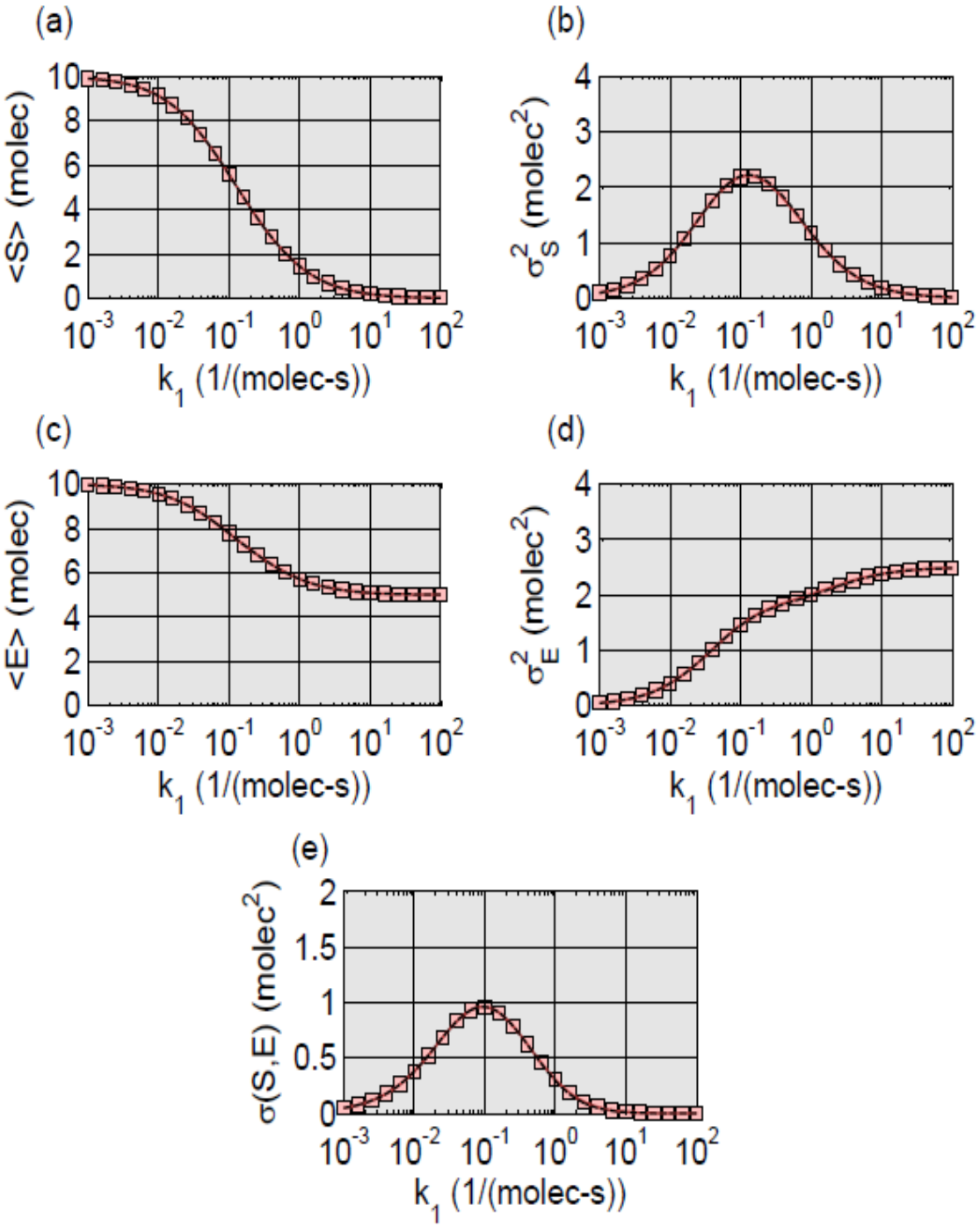

Figure S4.4 – Michaelis-Menten Steady-State Results across a range for k_4

The (a) mean S, (b) mean E, (c) variance for S, (d) variance for E, and (e) covariance for the steady-state Michaelis-Menten model with 2nd-order (dashed line) and 4th-order ZI-closure (solid line) compared to SSA results ($1 \cdot 10^6$ trajectories, squares). The value for k_1 is changed and all other parameters are set to one. The difference between 2nd and 4th-order closure is most evident in the covariance plot (e).

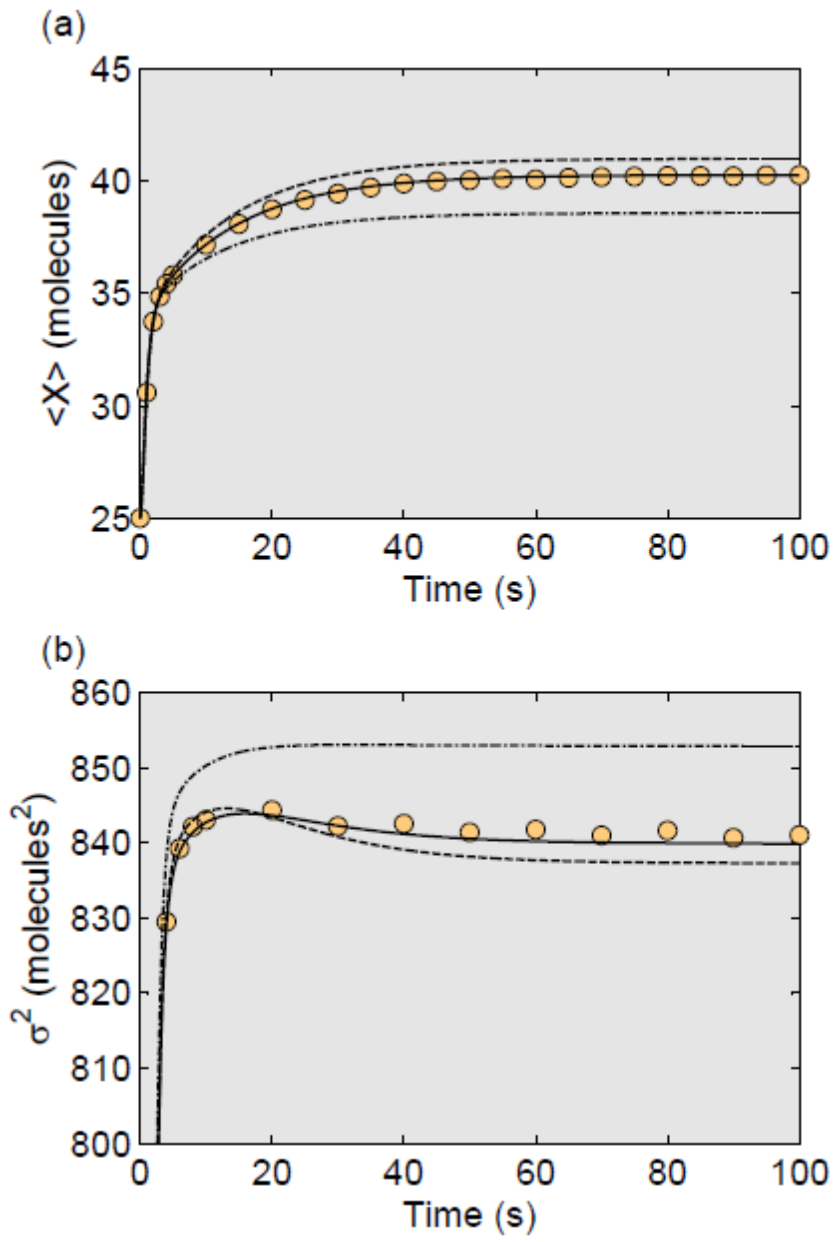

Figure S5.1 – Schlögl model 8th, 10th, and 12th-order ZI-closure results

The (a) mean and (b) variance results for 8th (dash-dot line), 10th (dashed line), and 12th-order ZI-closure (solid line) for the Schlögl model given the kinetic constants and initial conditions described in the body of the paper. The results are compared to SSA results ($1 \cdot 10^6$ trajectories, circles).

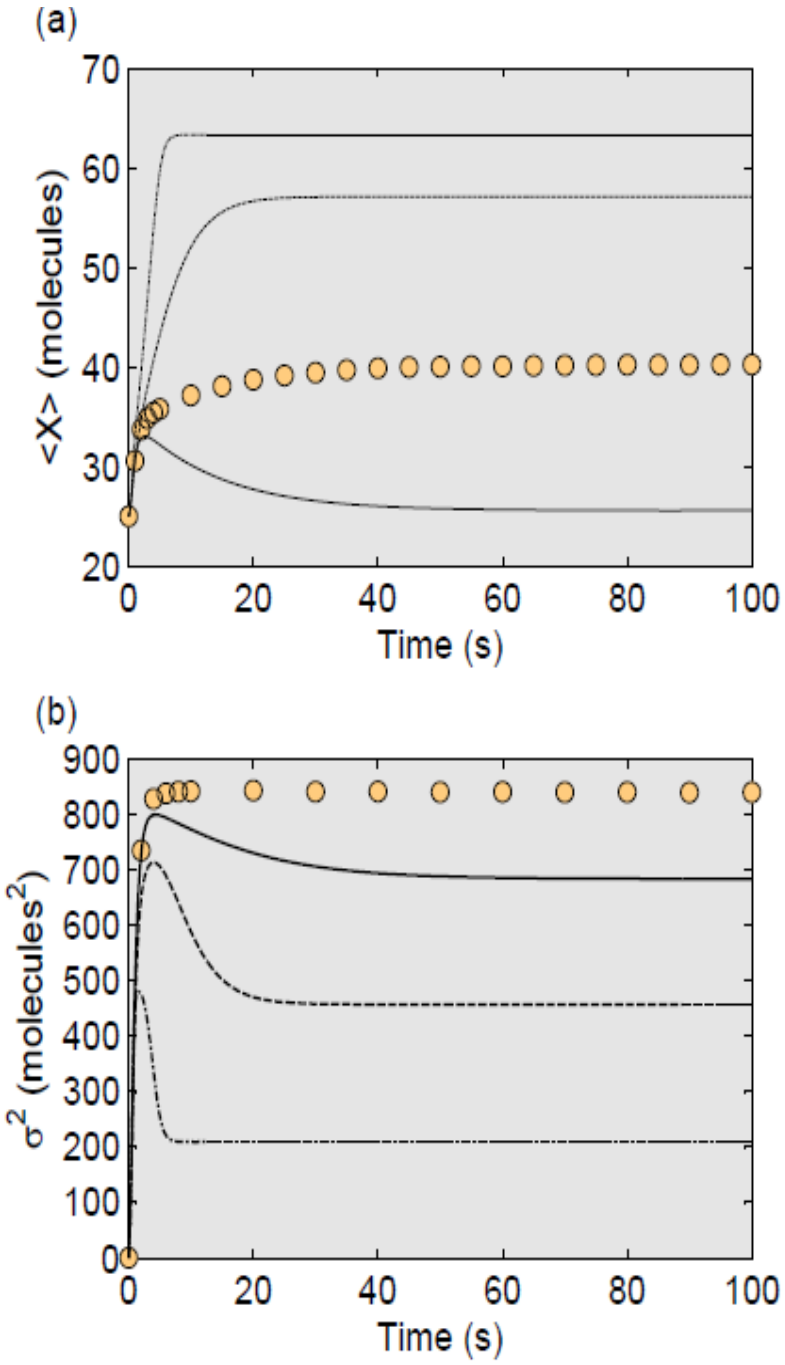

Figure S5.2 – Schlögl model 2nd, 4th, and 6th-order ZI-closure results

The (a) mean and (b) variance results for 2nd (dash-dot line), 4th (dashed line), and 6th-order ZI-closure (solid line) for the Schlögl model given the kinetic constants and initial conditions described in the body of the paper. The results are compared to SSA results ($1 \cdot 10^6$ trajectories, circles).

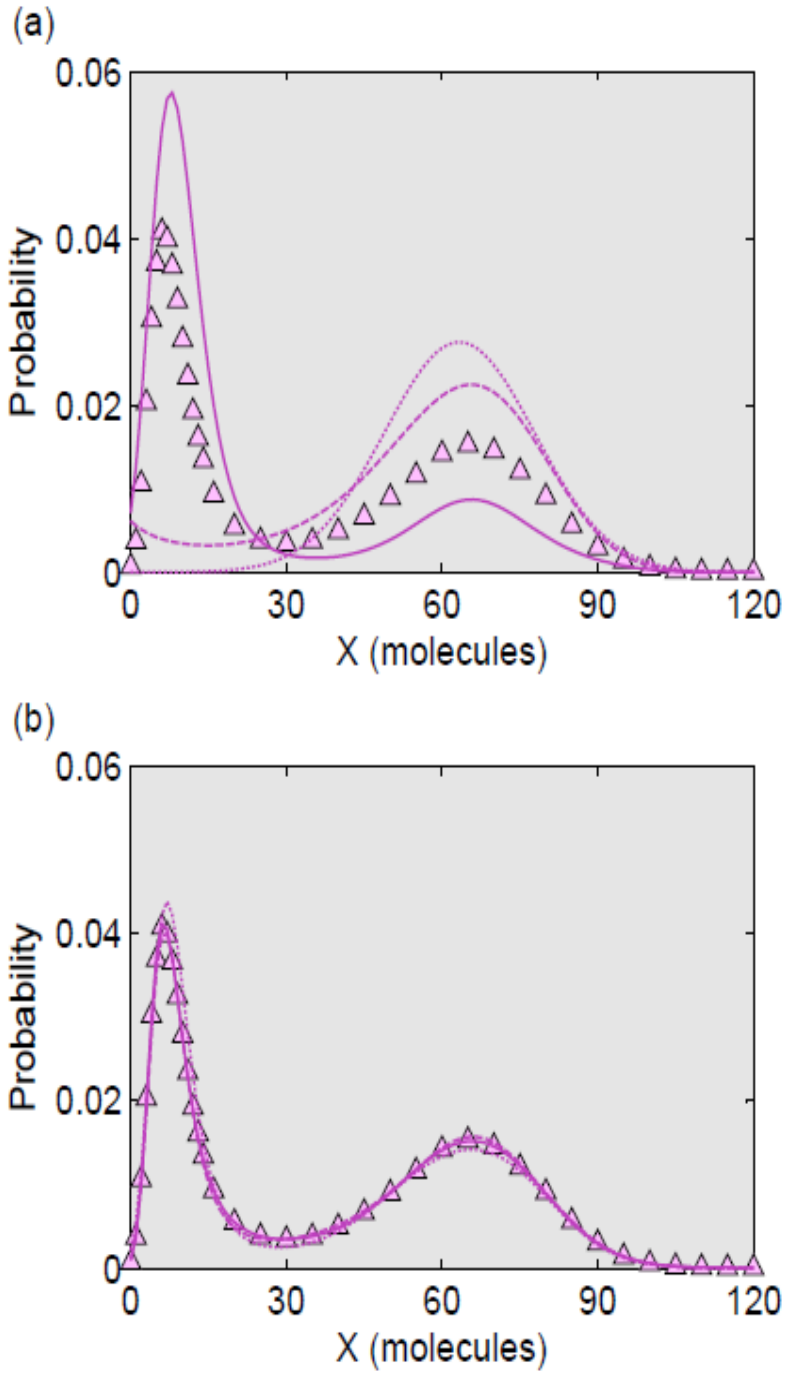

Figure S6 – Steady-State Probability Distributions for 2^{nd}, 4^{th}, 6^{th}, 8^{th}, 10^{th}, and 12^{th}-order ZI-closure

Steady-state distributions (sampled at t = 100 in the dynamic simulation, not from the steady-state optimization method) for (a) 2^{nd} (dotted line), 4^{th} (dashed line), and 6^{th}-order closure (solid line); (b) 8^{th} (dotted line), 10^{th} (dashed line), and 12^{th}-order closure (solid line). In all cases the results are compared to the SSA output ($1 \cdot 10^6$ trajectories, triangles). Kinetic constants and initial conditions described in the body of the paper.

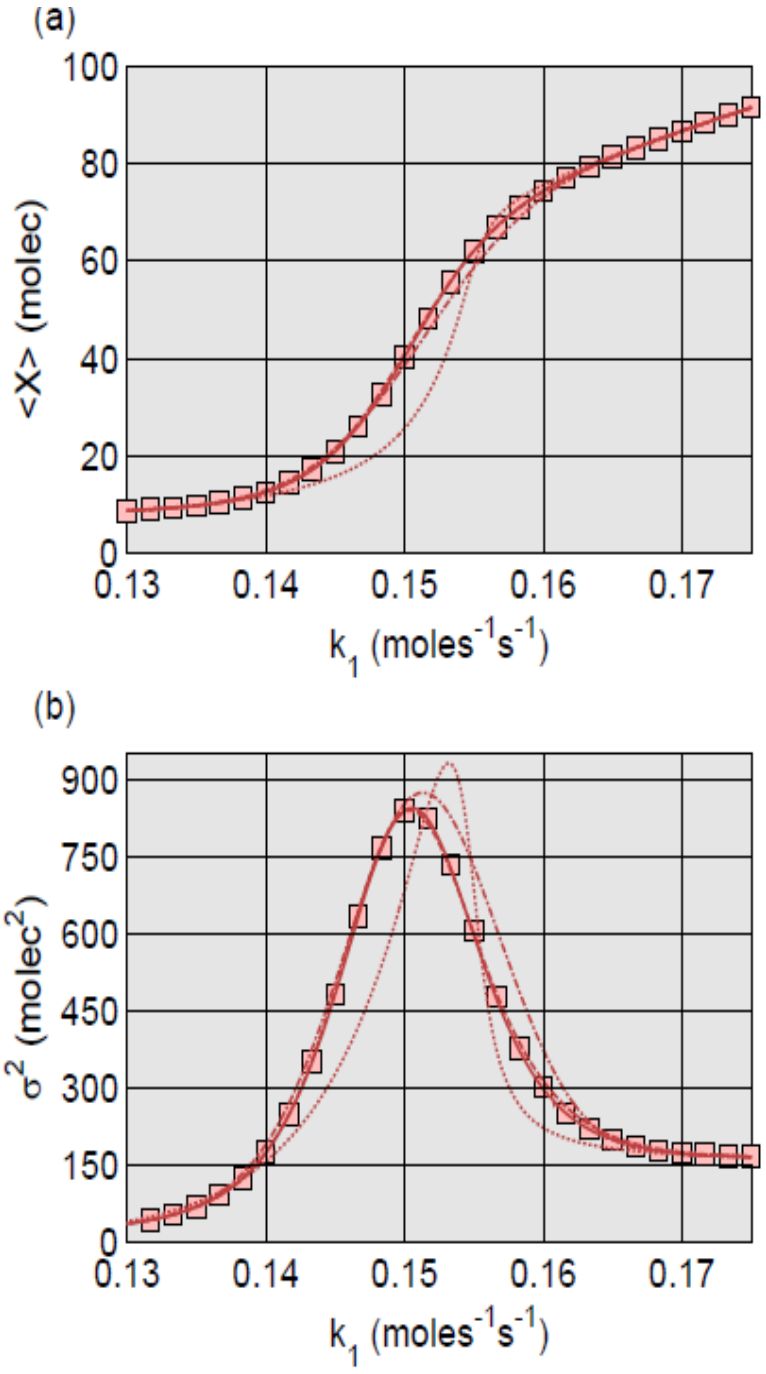

Figure S7.1– Steady-State Schlögl Model Results for k_1 with 6th, 8th, 10th, and 12th-order ZI-closure

The steady-state (a) mean and (b) variance results for the Schlogl model when k_1 is ranged (all other parameters held constant) for 6th (dotted line), 8th (dot-dash line), 10th (dashed line), and 12th-order ZI-closure (solid line). In all cases the SSA results ($1 \cdot 10^6$ trajectories, squares) are provided for comparison.

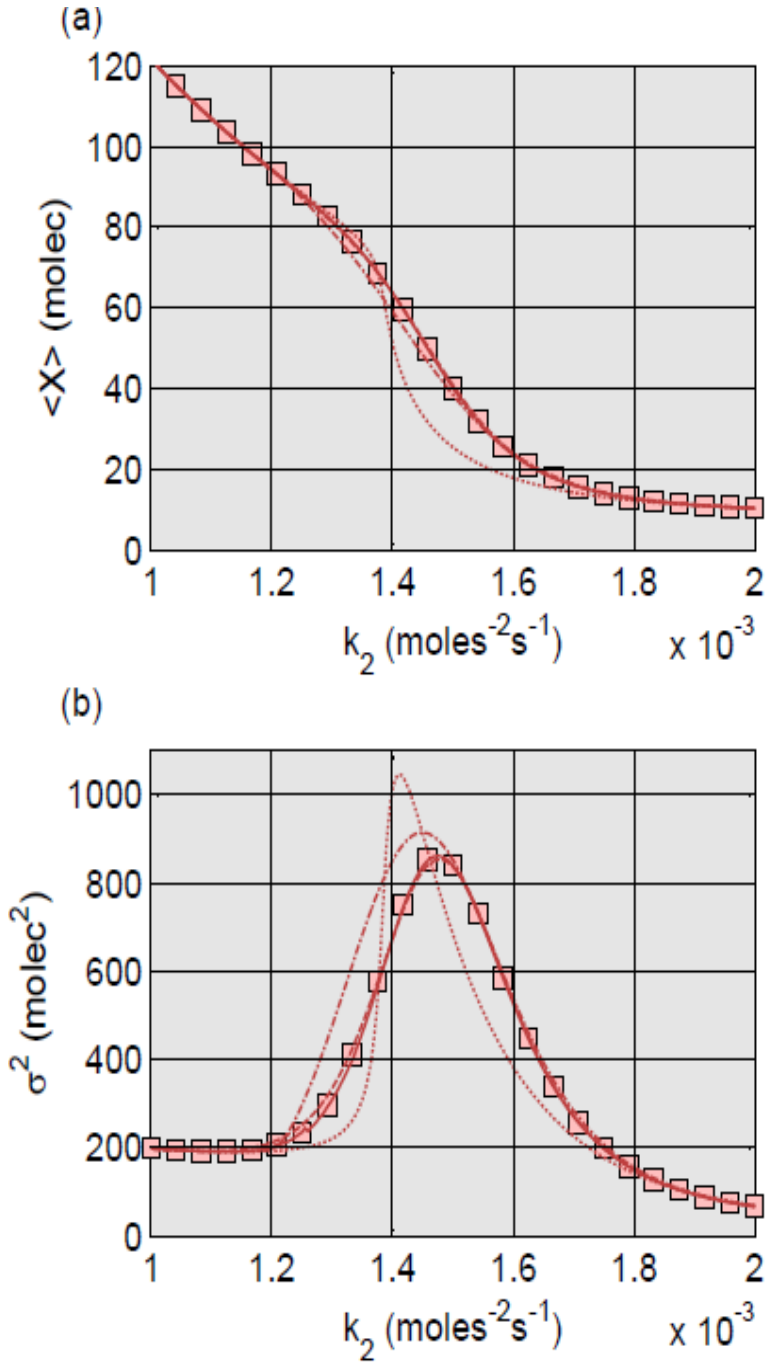

Figure S7.2 – Steady-State Schlögl Model Results for k_2 with 6th, 8th, 10th, and 12th-order ZI-closure

The steady-state (a) mean and (b) variance results for the Schlogl model when k_2 is ranged (all other parameters held constant) for 6th (dotted line), 8th (dot-dash line), 10th (dashed line), and 12th-order ZI-closure (solid line). In all cases the SSA results ($1 \cdot 10^6$ trajectories, squares) are provided for comparison.

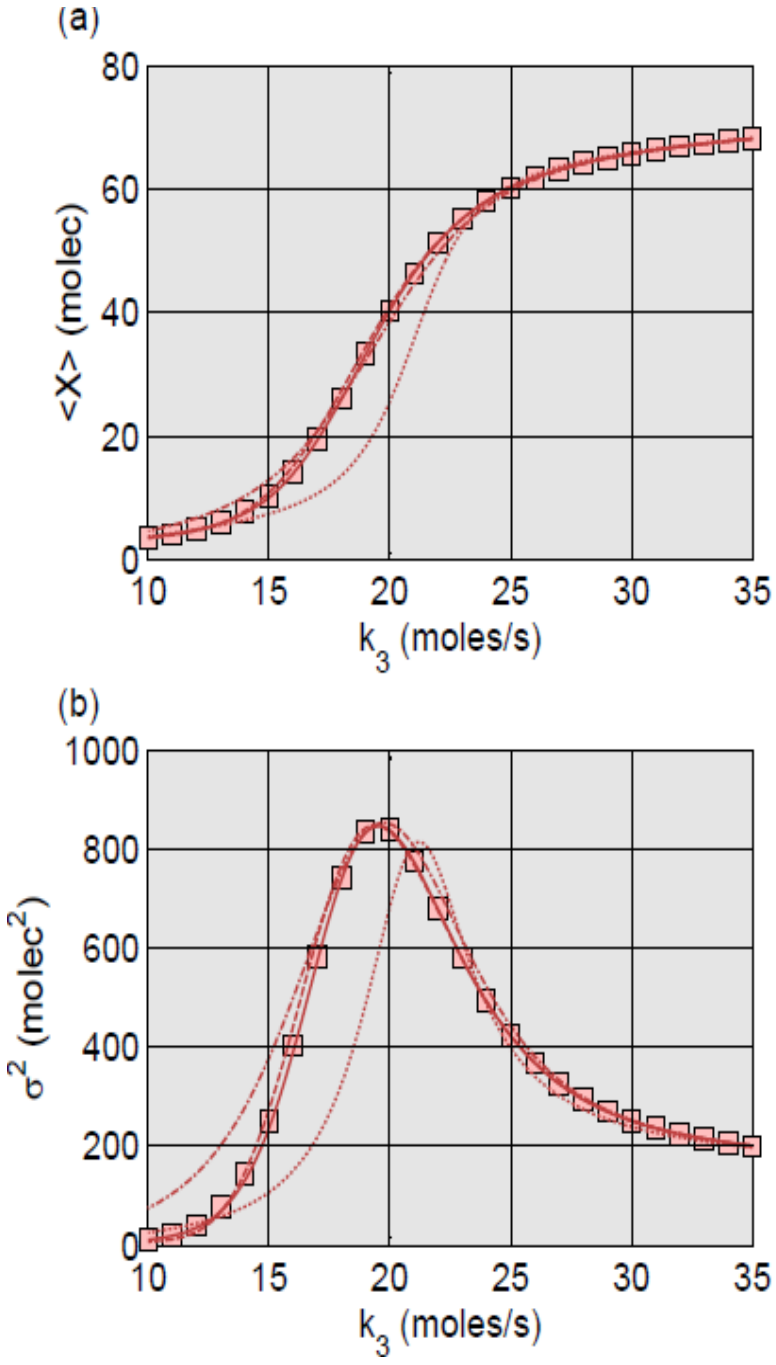

Figure S7.3 – Steady-State Schlögl Model Results for k_3 with 6th, 8th, 10th, and 12th-order ZI-closure

The steady-state (a) mean and (b) variance results for the Schlogl model when k_3 is ranged (all other parameters held constant) for 6th (dotted line), 8th (dot-dash line), 10th (dashed line), and 12th-order ZI-closure (solid line). In all cases the SSA results ($1 \cdot 10^6$ trajectories, squares) are provided for comparison.

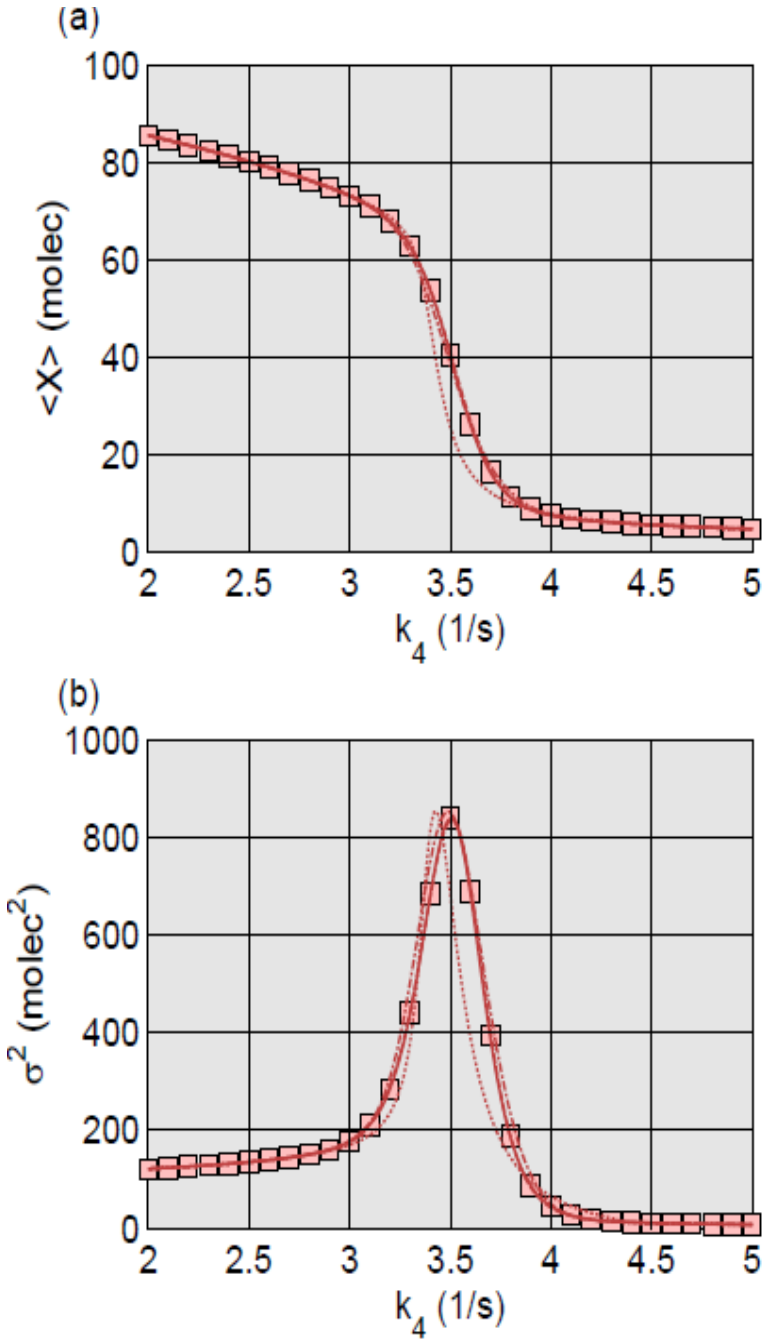

Figure S7.4 – Steady-State Schlögl Model Results for k_4 with 6th, 8th, 10th, and 12th-order ZI-closure

The steady-state (a) mean and (b) variance results for the Schlogl model when k_4 is ranged (all other parameters held constant) for 6th (dotted line), 8th (dot-dash line), 10th (dashed line), and 12th-order ZI-closure (solid line). In all cases the SSA results ($1 \cdot 10^6$ trajectories, squares) are provided for comparison.